Universidade de Aveiro
2014
Departamento de
Electrónica, Telecomunicações e Informática

André Gonçalo
Sabino Valente

DSCL and GSCL in ETSI M2M platform

DSCL e GSCL na plataforma ETSI M2M

Universidade de Aveiro Departamento de
Electrónica, Telecomunicações e Informática
2014

André Gonçalo
Sabino Valente

DSCL and GSCL in ETSI M2M platform
DSCL e GSCL na plataforma ETSI M2M

Dissertação apresentada à Universidade de Aveiro para cumprir os requisitos necessários para a obtenção do grau de Mestre em Engenharia Eletrónica e Telecomunicações, realizado sob a orientação científica do Professor Doutor João Paulo Barraca e Professor Doutor Diogo Gomes.

agradecimentos / acknowledgements

Os meus agradecimentos dirigem-se especialmente ao apoio incontestável e contínuo de todos os membros da minha família destacando o apoio moral e psicológico da minha Mãe, a quem dedico este curso. Agradeço a paciência infinita do meu Irmão Miguel e o seu despreocupado otimismo. Saliento a fortíssima motivação e coragem transmitida pela mulher da minha vida Maria, que esteve sempre comigo em todos momentos altos e baixos deste periodo da minha vida, algo que espero ter a possibilidade de retribuir da melhor maneira. Tambem queria agradecer a paciência e disponibilidade dos orientadores: João Paulo Barraca e Diogo Gomes, os quais certamente aconselharei a futuros alunos de dissertação de mestrado. Agradeço todas as dificuldades que enfrentei; não fosse por elas, eu não teria saído do lugar, mesmo as críticas construtivas e destrutivas me auxiliaram muito ao longo deste periodo de estudante universitário. Subscrevo-me como pequeno homem com grandes aspirações.

o júri / the jury

presidente / president	**Professor Doutor Osvaldo Manuel da Rocha Pacheco** Professor Auxiliar da Universidade de Aveiro
arguente / discussant	**Prof. Doutor José de Castro Ferreira** Professor Adjunto da Universidade de Aveiro
vogais / examiners committee	**Prof. Doutor João Paulo Silva Barraca** Professor assistente da Universidade de Aveiro

Keywords M2M, IoT, GSCL, DSCL,ETSI, Sensor Node

Abstract Machine to Machine (M2M) is a technology facing huge expansion in the coming years. One can claim that digital technological breakthroughs have changed modern society and the way it interacts with the surroundings. The decreasing prices of sensors/actuators have led to a completely new way of managing industry. The abundance of existing networks, has revolutionized the way a company can check, provide and regulate information about his product. A concept that we must take into account and is very often associated with M2M communications is the *Internet of Things*, the purpose of this approach is to connect billions of objects and take advantage of their functions as many times as intended. This dissertation will have its focus on developing components for an architecture node M2M. It will focus on certain critical components like the Device Service Capabilities Layer (DSCL), belonging to ETSI M2M architecture according to TS 120 690 standard. These components are traditionally called sensor nodes, in this document technical requirements and limitations are going to be faced aiming a final design.

Keywords M2M, IoT, GSCL, DSCL,ETSI, Sensor Node

Abstrato

M2M é uma tecnologia voltada para grande expansão nos próximos anos. Pode-se afirmar que os avanços tecnológicos digitais mudaram a sociedade moderna e a forma como esta interage com o ambiente. Os preços decrescentes de sensores/atuadores levaram a uma forma completamente nova de gestão da indústria. A abundância de redes existentes, revolucionou o modo como uma empresa pode verificar, fornecer e regular as informações sobre o seu produto. Um conceito que temos de levar em conta e é muito frequentemente associado com comunicações M2M é a *Internet das coisas*, o objetivo dessa abordagem é conectar bilhões de objetos e tirar proveito de suas funções varias vezes. Esta dissertação terá seu foco no desenvolvimento de um nó para uma arquitetura M2M. Esta irá se concentrar em determinados componentes críticos, como o DSCL, pertencente à arquitetura M2M do standard European Telecommunications Standards Institute (ETSI) de acordo com o padrão TS 120 690. Estes equipamentos são tradicionalmente chamados nós, neste documento os requisitos técnicos e as limitações vão ser confrontados com o objetivo de projetar um protótipo final.

Contents

List of Figures

List of Tables

Acronyms

6LoWPAN IPv6 over Low power Wireless Personal Area Networks

ADC Analogue to Digital Converter

API Application Programming Interface

CAN Controller Area Network

CO2 Carbon Dioxide

CoAP Constrained Application Protocol

CSMA/CA Carrier Sense Multiple Access with Collision Avoidance

DA Gateway Application

DSCL Device Service Capabilities Layer

DSSS Direct-Sequence Spread Spectrum

EEPROM Electrically Erasable Programmable Read-Only Memory

EPC Evolved Packet Core

ETSI European Telecommunications Standards Institute

EU European Union

GA Device Application

GA Gateway Access

GFSK Gaussian Frequency-Shift Keying

GPRS General packet radio service

GPS Global Positioning System

GSCL Gateway Service Capabilities Layer

GSM Groupe Special Mobile

GTS Guarantee Time Slot

HTTP HyperText Transfer Protocol

I2C Inter-Integrated Circuit

ICT Information and Communication Technologies

IEEE Institute of Electrical and Electronics Engineers

IET Institution of Engineering and Technology

IETF Internet Engineering Task Force

IMS IP Multimedia Subsystem

IoT Internet of Things

IoT-i Internet of Things Initiative

IP Internet Protocol

IPv6 Internet Protocol version 6

JSON JavaScript Object Notation

JXTA Juxtapose

LTE Long Term Evolution

M2M Machine to Machine

MAC Media Access Control

MCU Micro Controller Unit

MEMS Micro-Electro-Mechanical Systems

NFC Near Field Communication

NGC Network Generic Communication

NIP Network Interworking Protocol

NSCL Network Service Capabilities Layer

OSI Open Systems Interconnection

OTA Over The Air

PWM Pulse-width modulation

RAM Random Access Memory

RFID Radio-Frequency Identification

RISC Reduced Instruction Set Computer

SCL Service Capabilities Layer

SPI Serial Peripheral Interface

SRAM Static Random Access Memory

UDP User Datagram Protocol

USB Universal Serial Bus

WiMax Worldwide Interoperability for Microwave Access

WPAN Wireless Personal Area Network

WPAN Wireless Personal Area Network

WSN Wireless Sensor Network

Chapter 1

Introduction

The Internet of Things (IoT) is a growing concept in today's developed world, with increasing coverage of wireless networks the number of small connectible devices have expanded exponentially, bringing day-to-day accessories to the web. These devices connectivity gives the IoT great amounts of data motivating the evolution of M2M. The technological *boom* brought to the surface problems such as interoperability, a significant number of solutions were developed by manufacturers in the past years. These corporations though only about developing their technology and creating their own standard [1], hindering communication between different solutions. In view of the IoT and M2M, the concept of heterogeneity is a problem that limits the expansion of the technology, in fact interaction should be encouraged and not limited. The necessity for a M2M standard such as the one proposed by ETSI rises, the organization already provides documents introducing the definition of the requirements [2] and the implementation of functional architecture [3] and interfaces [4] necessary. Despite the effort, ETSI is not considered the final standard solution used globally, but is solid start. In M2M the human interaction is limited as much as possible, meaning that communication and interaction between machines occurs without human intervention. The *travelling* data could have been originated by a smart phone or from any other device able to connect to a network, information can be processed immediately or stored for future profiteering. The importance of these existing devices is crucial for the existence of information to be sent, more devices with more capabilities means more information that entities can access. Developing a functional, smart, autonomous and interoperable device together with an universal platform, or standard, could mean a huge step towards and a greater expansion of the IoT. As already mentioned ETSI is not considered the global standard, but it is an initiative that presents a concrete solution and motivates the development of devices to integrate this proposal.

1.1 Motivation

The urge to create new market opportunities while providing a competitive advantage for enterprise users is a key issue in today's economy. Thus all boils down to competition in a range of areas including automotive and logistics, utility, retail, healthcare, agriculture and public safety between others.

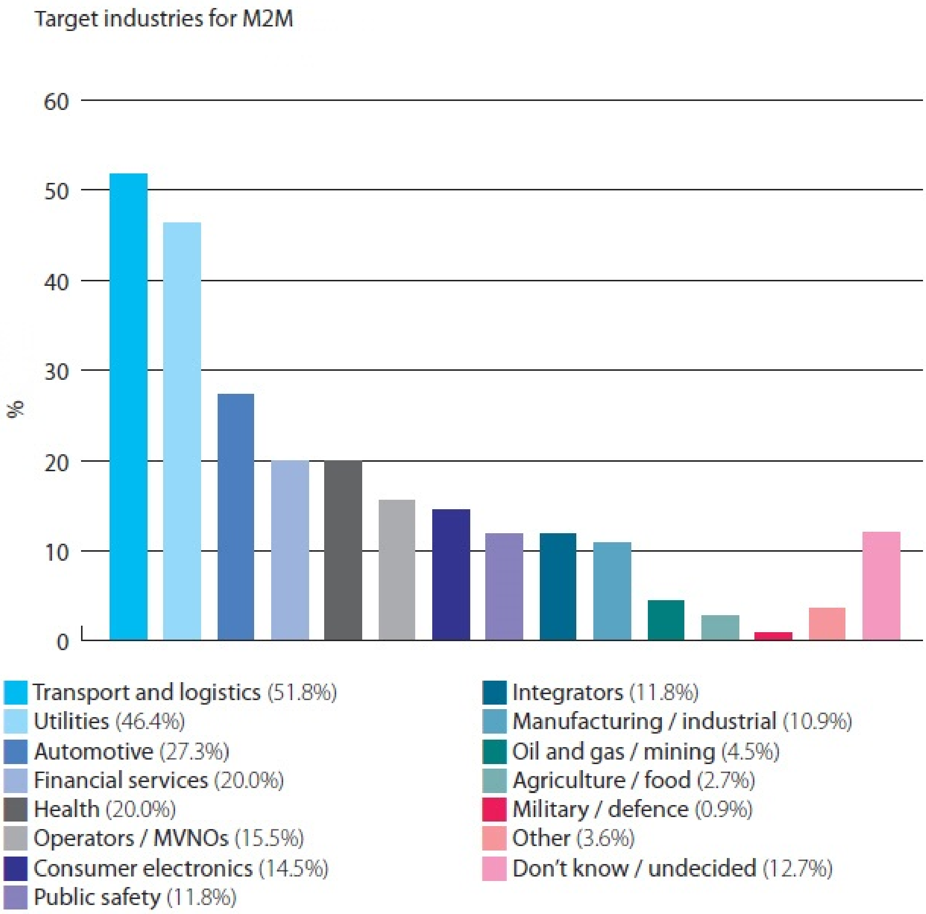

Figure 1.1: Informa-SAP M2M Communication Survey, 2012[1]

M2M technology focus on connections, but whether it is 10 billion, 20 billion, or 40 billion, there is no question that the number is huge, from the IoT point of view these significant amount of connections represent an opportunity. Due to continuous technology evolution and price decrease, enterprises will increasingly turn to M2M to lower operating costs and differentiate their brands, creating revenue opportunities. The development of a device that could satisfy both the technical requirements, but also the economic struggle of the competition has become a necessity, in fact the M2M communication is motivated by the continuous increasing integration of data within information systems, leading to a more strategic use of it. The emergence of mass data requirements for cross-sector service opportunities also stimulates the development and expansion of this M2M devices. As an important part of the final architecture the device developed and presented in this dissertation is the first step towards a functional, intelligent and profitable technology. In fact, standing himself at a lower layer, is responsible for data collection. This *observer* of the physical world should be able to fulfil every task in the most reliable and profitable way, so that the rest of the layers above the node are already one step beyond and cost-effective compared to other technologies. It is also foreseen that the development of the device must include interoperability referring to ETSI M2M standard. In a world composed of networks and sub-networks it is expected that standards and technical regulations exist allowing the creation of adaptable, scalable and heterogeneous connections. Therefore, this Dissertation objectives are framed within the provisioning of such a device capable of satisfying hardware, software and standard restrictions.

1.2 Objective

The focus of this document is centred in the development of a device able to collect information from the surroundings and sending data via a communication protocol through a wireless connection. The technology must also satisfy some key requirements of M2M ETSI standard. The node must fulfil critical hardware requirements faced and defined through the document, so that it becomes possible to create a smart functional device.

[1]http://images.tmcnet.com/tmc/misc/articles/Image/2012/M2M2.jpg

The node was designed to comply pre-established specifications: low battery consumption, easy and fast sensor expansion/substitution and wireless communication. Additionally, this document provides an analysis of existing projects based on M2M ETSI standard and a range of similar device technologies. Since the document describes briefly the ETSI layers, it is possible to have a perception of system hierarchy and understand the presented scenarios and projects.

1.3 Structure

This dissertation document is divided and organized in 7 chapters, being the first chapter the Introduction and the final one the Conclusion.The remaining chapters are:

- Chapter 2: provides information of the state of the art. The key concepts described are M2M, IoT, Wireless Sensor Network (WSN) technologies.
- Chapter 3: general analysis of ETSI M2M standard, together a brief explanation of the technical approach and the architecture; similar projects are also exposed
- Chapter 4: describes the node architecture of a DSCL together with the requirements that had to be fulfilled
- Chapter 5: starts by explaining how the design was faced and presents a range of components to be used within the development of the node, finally it describes real applications of similar devices
- Chapter 6: presents the developed device, the tests performed with the top layer, a brief data analysis finishing with a discussion

Chapter 2

State of the Art

2.1 History and Technological evolution

M2M has been applied in many areas and exists in many forms, predominantly in cellular communication applications. Machine-to-machine, machine-to-man or man-to-machine communications are expected to grow quickly in the next 10 years, reaching the 50 billion available devices with broadband connections access by the year of 2020. In 1995 Siemens invested in the first M2M department, and by doing so they were able to establish the fist machine-to-machine communication over a wireless network. These advances led to the creation of an independent separated business unit inside Siemens in 2000 called "Wireless Modules". Around 1997, with the affirmation of wireless technology, the M2M devices and modules were designed for specific tasks such as automotive telematics. In our days the complex and sophistication of M2M devices is extremely high, and include many functionalities to improve applications and expand the usage of this machines as far as imagination can go.

2.2 Internet of Things

2.2.1 Current applications and examples

In our present days the IoT is currently applied in a variety of cases such as: homes, businesses and cars. A common application is, for example, in homes where a high number of connectible objects exist, including: lights, thermostats, key tracking devices and many others. Having the power to control these devices remotely can lead the owner to huge resource savings in gas, water and electricity. Business applications of IoT include temperature and light control, but also applications controlling malfunctioning machines and vending machines, where the device could signal a defective behaviour to the administrator. These two IoT applications listed above are very simple ways of using this technology, in fact it could be used to his full potential connecting multiple devices and make them communicate and interact with each other. A possible simple scenario of an application using an email account connected to an alarm clock, which is connected to a toaster and/or a coffee machine. For instance: a worker is notified via email that his job will start one hour earlier tomorrow, with the use of IoT, the alarm clock could set to a new wake up hour and by ringing could communicate to the coffee machine to start her process. This simple idea is one of the numerous applications, it is just necessary to bring the physical objects on-line.

2.2.2 Future developments

IoT can be identified as one of the main areas in the future of the internet [5], in fact this area presents a huge potential, making promise of impact in economy and society; as shown in Figure 2.1 by the evolution of the number of connections. While looking into the next stages of this technology there are many opportunities and challenges to be faced including privacy concerns, security, costs, standards and regulations. For example the Internet of Things Initiative (IoT-i) [6] coordinated action in the European Commission has settle three main objectives to achieve in the next years:

- Creating a joint strategic and technical vision for the IoT in Europe that encompasses the currently fragmented sectors of the IoT domain

- Contribute to the creation of an economically sustainable and socially acceptable environment in Europe for IoT technologies and respective research and development activities

- Creating an environment that favours international adoption of European IoT technology

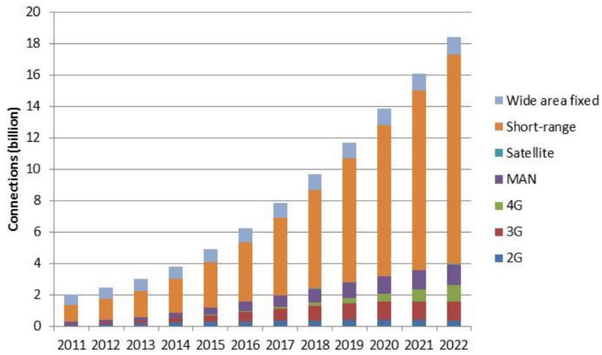

Figure 2.1: Global M2M connections 2011-22 by technology based in [7]

2.3 Machine to Machine

2.3.1 Meaning and concept

M2M communication refers to intercommunication between two or more devices without human interaction, using M2M communication protocols. As mentioned previously M2M communications are used for a great number of applications connecting any type of electrical, electronic machine and mechanic over a large variety of existing networks. Applications are found in: wireless healthcare, remote data collection, remote control, remote payment, remote monitoring,telemetry, telematics.

2.3.2 Hardware: Device Examples

Before describing device examples common characteristics found in each device are going to be defined, such as:

- **Power:** In almost every application of these devices batteries can be found as power sources, since they can be deployed in unreachable places and are expected to work for long periods of time without any maintenance.
- **Memory:** It can be Flash,Electrically Erasable Programmable Read-Only Memory (EEPROM)or Random Access Memory (RAM), application data or user related data , program memory or data memory. RAM is a fast access memory where items can be read and written almost in the same amount of time, but the memory is volatile and the content of it is lost when the device is tuned off. Flash is a non-volatile type of memory which can be erased and reprogrammed electrically, this type of memory is an evolution of EEPROM. The difference between EEPROM and Flash is that the first one has to be completely erased to be rewritten, Flash can be erased in blocks.
- **Controller:** Usually this part of the device is a micro-controller responsible for the device information, data and instructions processes.
- **Sensor/Actuator:** These components are responsible for the interaction of the device with the physical world, using a Digital-to-Analogue or Analogue-to-Digital converter. A device can contain more than one of these components, normally incorporated in the micro controller.
- **Transceiver:** Definitely one of the components that has higher battery consumption. Using technologies such as WiFi, ZigBee, Bluetooth; these components allow communication with other devices/machines [8].

Now that the notion of the components composing the device have been presented, it is possible to proceed to an exposure of existing devices and platforms directly connected to the dissertation's objective.

2.3.2.1 Sun SPOT

Sun SPOT [9] stands for Sun Programmable Object Technology, are small devices with wireless capabilities powered by a battery developed by Sun Labs. These devices are used in a large number of applications like robotics, monitoring, tracking and many others.

These devices have a Java virtual machine, called *Squawk*, this platform can run different applications, it is important to highlight that these devices support Over The Air (OTA) programming. Stackable boards can include specific sensors and/or actuators, depending on the goal. Since the device is battery-powered, it has the possibility to be duty cycled to permit autonomy that lasts for mouths without a power recharge. A more detailed look at this technology, for example the hardware specifications: each Sun SPOT has a 400MHz 32-bit ARM920T core processor with 1M RAM and 8M Flash, it communicates via an integrated antenna with a 2.4GHz radio, which is a TI CC2420 (formerly ChipCon) and is IEEE 802.15.4 compliant. The Sun SPOT device is powered by a 3.7V rechargeable, 750 mAh lithium-ion battery; this permits a 7 hours operation with CPU and radio active, the operational time can be extended by turning the device to sleep mode. The dimensions of the device are contained, in fact it measures 41 x 23 x 70 mm and it weights 54 grams.

2.3.2.2 Arduino

This technology is an open-source physical computing platform build on a basic I/O board. Arduino [10] can be used for many purposes, and one of them is a stand-alone interactive object, in fact with the right gadgets it can sense the environment and affect the surrounding by controlling lights, heaters and other actuators. The technology board is cheap and robust it's based on an ATmega micro-controller; the CPU speed, number of analog In/Out ports, the quantity of: EEPROM, Static Random Access Memory (SRAM) and Flash memory vary with the model that is necessary. For example the Arduino Uno, one of the basic models, has an ATmega328 micro-controller with a clock speed of 16MHz, an operating voltage of 5V, 14 digital I/O pins (of which 6 provide PWM output), 6 analog pins, 32 KByte flash memory of which 0.5 KByte used by the bootloader, 2 KByte SRAM and 1KByte of EEPROM. It is possible to find interesting gadgets for the Arduino, for example a long range antenna witch can reach up to 1 Km with up to 2Mbps, high efficient GFSK modulation. The dimensions of the Uno model are 75 x 53 x 15 mm, with a weight of approximately 30 grams. The Arduino family is composed of many devices to satisfy the requirements of a possible user, they can vary in physical characteristics, processing capacity, number of pins, communication interfaces, memory, between various features available. The table bellow shows the comparison between some Arduino technology that could be used for the purpose of this dissertation.

Name	Processor	Operating Voltage /Input Voltage	CPU Speed [MHz]	Analog In/Out	Digital IO/PWM	EEPROM [KB]	SRAM [KB]	Flash [KB]	USB	UART
Micro	ATmega32u4	5 V/7-12 V	up to 16	12	20/7	1	2.5	32	Micro	1
Mini	ATmega328	5 V/7-9 V	up to 16	8	14/6	1	2	32	-	-
Nano	ATmega168/328	5 V/7-9 V	up to 16	8	14/6	0.512/1	1/2	16/32	Mini-B	1

Table 2.1: Arduino model comparison

2.3.2.3 Waspmote

Waspmote [11] is an open source wireless sensor platform, designed for low consumption modes allowing the sensor nodes to work autonomous; based on a modular architecture in a way that the device is upgradable through proper sockets, allowing to install the needed modules(some sensors are available already on board, as default). Taking a look at this technology in comparison with the other analysed, it is possible to rapidly conclude that Waspmotes have a more humble specifications, in other hand it furnish a higher amount of communications solutions and topology implementation. Looking at the general characteristics Waspmote runs with, 14Mhz ATmega microcontroller, 8 KByte SRAM, 4 KByte EEPROM, 128 Flash and expandable memory up to 2 GByte with SD card; the device provides 7 analogue Inputs, 8 digital I/O, 2 UART, 1 Inter-Integrated Circuit (I2C), 1 Serial Peripheral Interface (SPI), 1 Universal Serial Bus (USB) port. The battery is USB charged and works between 3.3V and 4.2V, where the device presents a consumption in *on-mode* of 15mA and in *sleep-mode* of 55uA. As already mentioned this technology provides the user with a higher variety of wireless interfaces, such as:

- 802.15.4/ZigBee
- WiFi
- IPv6 over Low power Wireless Personal Area Networks (6LoWPAN)/Internet Protocol version 6 (IPv6) Radio
- 3G+Global Positioning System (GPS)

- Groupe Special Mobile (GSM)/General packet radio service (GPRS)
- Bluetooth
- Radio-Frequency Identification (RFID)/Near Field Communication (NFC)

As in Sun SPOT, Waspmote devices, allow OTA programming, which can be very useful when the sensors and/or actuators are out of human reach.

2.3.2.4 mbed

mbed [12] is a platform for developing devices that are based on ARM Cortex-M 32-bit micro controllers. It is designed to provide a highly productive solution for prototyping and product development, with a focus on connected Internet of Things devices. ARM is the developer of this technology, which is used by tens of thousands of professional developers to create intelligent products that take advantage of the power of modern micro controllers and connectivity. The mbed micro controller, the mbed NXP LPC17680, has an ARM Cortex M3 core, running at 96 MHz, with 512 KB flash, 64 KB RAM, including interfaces such as: Ethernet, USB Device, Controller Area Network (CAN), SPI, I2C and other I/O. The mbed software development kit provides C and C++ software platform and tools for creating firmware that runs on the devices, consisting of core libraries.

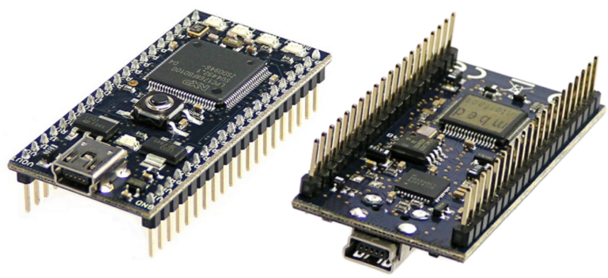

Figure 2.2: mbed device example[1]

[1]http://elmicro.com/site/assets/files/1381/mbed-nxp-lpc1768.png

2.3.3 First approaches and solutions

2.3.3.1 Vertical

At first the M2M systems had a low complexity and were composed only of a network of sensors/actuators and services/applications, and it was only required to program services and how they would interact with the network, due to the simplicity of the system. These systems were referred as vertical, and present some dependencies that weaken the system. In fact to perform the necessary operations the services have knowledge of the way they must address devices and the proper protocols to interact with them. The services are in full responsibility of interactions between the networks and themselves causing some problems when the number of networks increase. To ensure low level interactions with the device gateways have to be implemented by the manufacturer, leading to a slow development of services. Figure 2.3 shows an example of vertical system, where the applications that allow communication with the devices have to be projected and implemented by all services. Leading to one of the main problems, it is not possible to implement one communication module for all the services, it must be implemented a different module for each service necessary. It could happen that a new independent network has to be created and configured, in fact the non existence of re-utilization of communication mechanisms can lead to the appearance of unnecessary and redundant networks. Obviously, in vertical implementations, interoperability can be very hard to achieve, since each manufacturer will use their own protocols and specifications to implement their network of devices and services. Vertical systems are appropriate for simple implementations since they present easy development and rapid implementation of their solution.

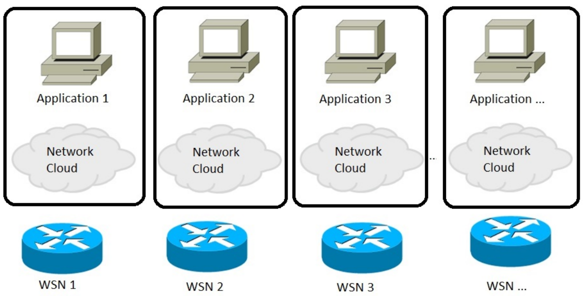

Figure 2.3: Vertical solution

2.3.3.2 Vertical vs Horizontal

Figure 2.4 shows the main difference between vertical and horizontal systems, it can be immediately noticed that horizontal architecture does not need one full infrastructure for each application, as in the one presented in section 2.3.3.1. Due to vertical system limitations, like scalability and interoperability, it was necessary to evolve the system turning it into a new one, more flexible and that could allow the integration of new networks of devices and systems. That justifies the appearance of horizontal systems that allow re-utilization of already developed infrastructures, presenting bigger scalability since the services are no longer responsible for managing the interactions between networks. IoT is based on this horizontal systems, that allow interaction between nodes in the network, integrating different networks of devices and systems.

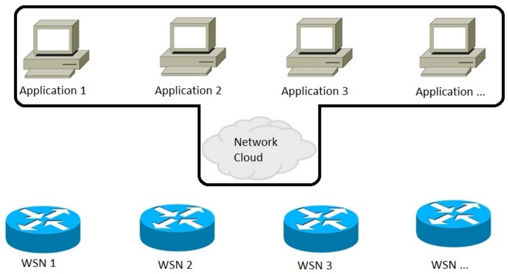

Figure 2.4: Horizontal solution

2.3.4 Framework requirements

The framework requirements are critical factors that need to be considered, the following topics itemize these requirements [13]:

- **Heterogeneity:** Heterogeneity refers to the ability that a network has to agglomerate different protocols and technologies. Taking into account that the main objective of a network is to permit communication between ends. An heterogeneous network has to work independently from the technology, specifications or protocol of the manufacturer of the machine that is connected, for instance a personal computer and an Apple machine.

- **Scalability:** In simple words a system performance increases, after new hardware is installed, in a proportional way compared to the capacity added can be called a scalable system. In other words a network has to keep up her performance even if the system increases size considerably.

- **Adaptability:** It is the ability that a system have to adapt efficiently to the sudden change of circumstances in a network. A functional network system should be able to deal easily, for example, with the addition or removal of a device/machine from itself, without compromising the entire structure.

- **Management:** Management can be seen as a key word when it comes to networks, in fact the simple action of giving authentication or authorization to a device can ensure him access to control entities and to consult resources. Also a smart management of information flow can easily influence the characteristics of the system and by doing so define quality of service and quality of information in the network.

- **information processing:** refers to information management and interpretation, there is a need for the existence of communication protocols allowing correct data exchange and interpretation. For instance a communication protocol needs to exchange a sequence of bits, these bits include the data read from a sensor, for example. The correct information processing comes when both ends are able to distinguish valid data from the necessary protocol bits, in other words both sides must *speak the same language*.

2.3.5 Communication Protocols and standards

2.3.5.1 Application Layer

CoAP stands for Constrained Application Protocol and is a software protocol that allows simple devices to communicate between each other over the internet. This protocol works in the application layer and it was projected to translate to HyperText Transfer Protocol (HTTP) to facilitate his integration with the web. Since it was designed for low consumption and small devices, Constrained Application Protocol (CoAP) seems to be perfect in M2M and IoT applications. In fact among the protocol characteristics there is multicast support and low overhead, which are welcome *in this world*, where efficiency is imperial. CoAP uses two types of message types based on a binary header format, the request message and the response message, this messages are the interaction model used between application endpoints; also important to mention that this protocol can run almost in all devices that support the User Datagram Protocol (UDP) features. Some of the key features of CoAP:

- Simple caching based on Time To Leave (TTL)
- Simple subscription for a resource, and resulting push notifications
- Content-type support
- Low header overhead and parsing complexity.
- Support for the discovery of resources provided by known CoAP services

2.3.5.2 ZigBee

ZigBee [14] is a standard wireless communication protocol that targets low-cost, low-power wireless sensor/actuator technology. ZigBee networks were first projected around 1999, this protocol technology can be found in different areas of the market, such as: commercial building management, consumer electronics, energy management, health care and fitness, home management, retail management and telecommunications. Based generically on a wireless mesh technology to maximise range and reliability, in Europe, ZigBee works on 868MHz (1 channel) radio band frequency. In other parts of the world like USA and globally, the protocol, operates respectively on 915MHz (10 channels) and 2.4GHz (16 channels). Working on the named frequencies data transmissions rates can reach speeds of 250 Kbits/sec at 2.4GHz.

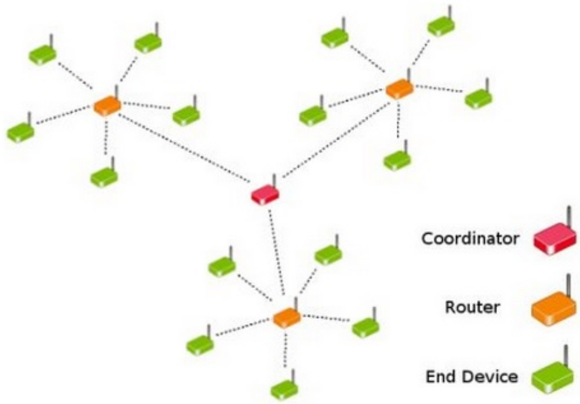

Figure 2.5: Example of a mesh network implemented using ZigBee [15]

Transmission distances can vary depending on the power output of the device, environmental conditions, buildings interiors and geographic topology; reaching distances up to 7000 meters in mesh topology measured in the best radio frequency conditions. ZigBee design is based, as defined in IEEE standard 802.15.4 [14], on the media access control and physical layer. The protocol, completes the standard by adding to the basic IEEE 802.15.4, four components as shown in Figure 2.6: network layer, application layer, ZigBee device objects; and completes the assembled with application objects designed to customize the technology. As already mentioned in this section, ZigBee, allows a really low consumption of the device battery, this happens because the network nodes can sleep most of the time, also nodes can awake from sleep mode in around 30ms ; giving the battery a long life.

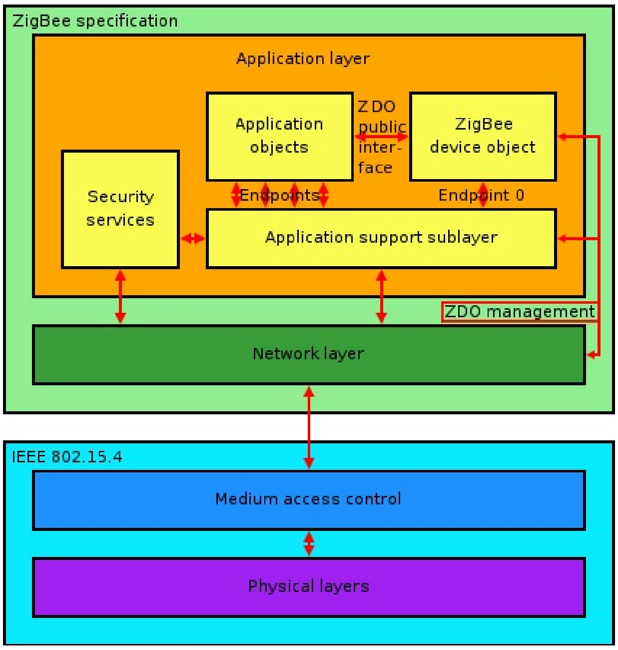

Figure 2.6: ZigBee protocol stack [16]

2.3.5.3 IEEE standard 802.15.4

[17] This standard focuses on low-speed, low-cost communication between devices in wireless network lower layers, as shown in Figure 2.7, it favours embedded devices with 10 meter communication range up to 100kbit/s in the latest versions. Very important features are included in this standard like real-time suitability, using reservation of guaranteed time slots; and collision avoidance through Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), also important to know that devices using this norm have three frequency band to operate on. Figure 2.7 shows us the two components of IEEE 802.15.4 standard: medium access control and physical layer. The first enables the transmission of Media Access Control (MAC) frames through the physical channel, it also offers a managing interface to control the access to the physical layer. It also controls frame validation, guarantees time slots and handles node associations. The physical layer is the lower layer in the Open Systems Interconnection (OSI) model, this layer provides a data transmission service and a management entity to maintain a database of information on related personal area networks.

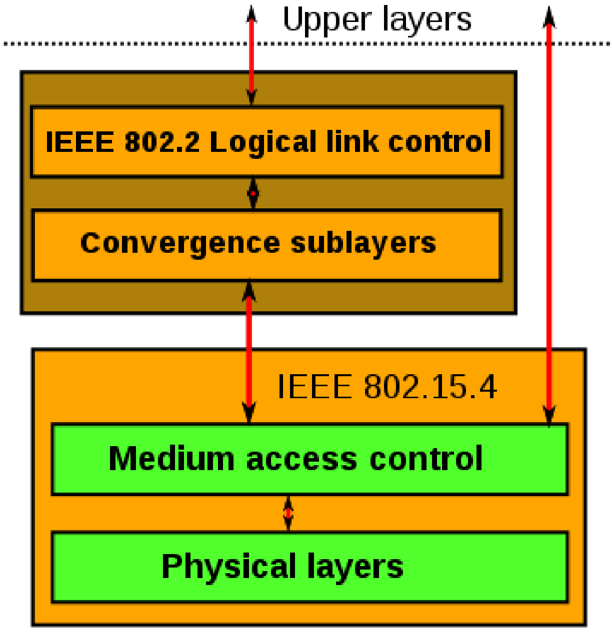

Figure 2.7: 802.15.4 architecture [16]

It is important to deepen this standard and understand why brings benefits against noise and interference, since is one of the basic communication standards used in M2M communication. 802.15.4 uses Direct-Sequence Spread Spectrum (DSSS), a modulation technique. DSSS is used to modulate the information before sending it to the physical layer, this process occupies more bandwidth but it uses lower spectral power density for each signal, by doing so it causes less interference in the frequency bands and improves signal to noise ratio in the receiver, as shown in Figure 2.8.

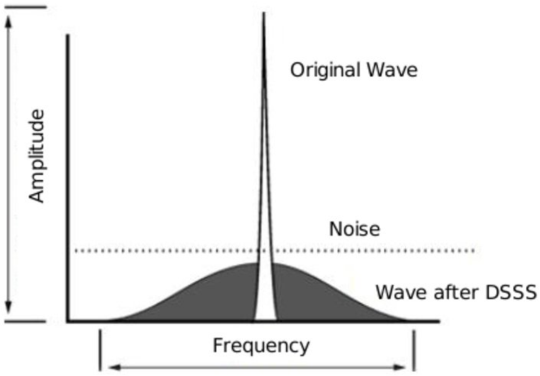

Figure 2.8: 802.15.4 spectrum analysis[2]

To avoid that all the nodes start emitting at the same time, leading to interference, this standard uses two techniques: CSMA/CA and Guarantee Time Slot (GTS). In CSMA/CA each node listens the medium prior to transmit. If the energy found higher of a specific level of a node, then the transceiver waits during a random time and tries again. In GTS the systems uses a centralized node (or coordinator node) which gives slots of time to each node so that any knows when they have to transmit.

2.3.5.4 HTTP

Stands for Hypertext Transfer Protocol [18], this application protocol was designed to satisfy the need of distributed and collaborative systems, commonly used today as base of the world wide web. Using logical links in his texts, the HTTP permits travelling between nodes and is based on a request-response protocol between client and server, permitting intermediate network elements to improve or enable communications between them. Methods are defined in HTTP to indicate the desired action to be performed on the identified resource.

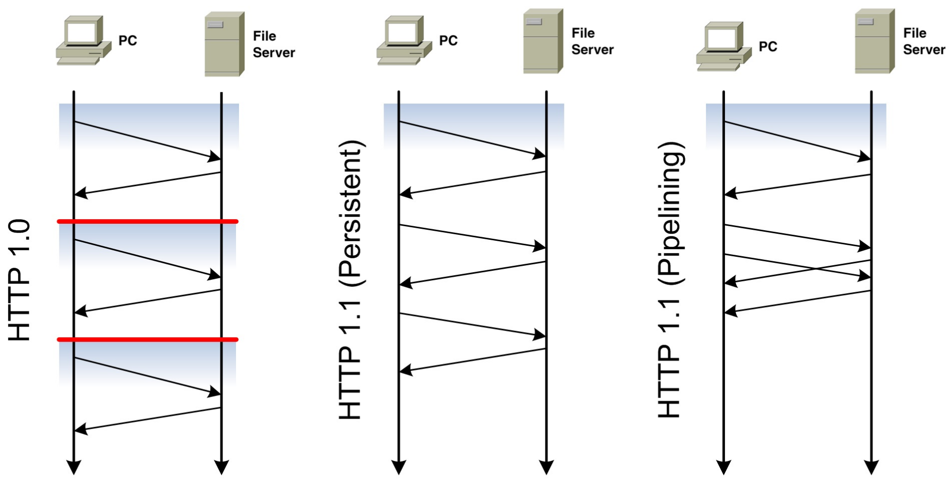

Figure 2.9: Comparison of HTTP evolution and graphical interpretation[3]

[2]https://www.tapr.org/images/ssfig1.gif

[3]http://dret.net/lectures/web-spring11/img/http-phttp-pipelining.png

2.4 Existing Projects

2.4.1 WattECO

WattECO [19] provides energy harvesting with Internet Protocol (IP) sensors for a smart grid or a smart building. This project works with IP based sensors implemented through 6LoWPAN [20] technology standardized by the Internet Engineering Task Force (IETF), expanding IP network to low-power automation applications. Their devices work without battery and are bidirectional radio based. The protocol used is 802.15.4 on 868MHz or 2.4GHz, adapted for complex applications. The company provides hardware and software, and they implement the technology over a 6LoWPAN stack implementation over a wide variety of transceivers and physical layers.

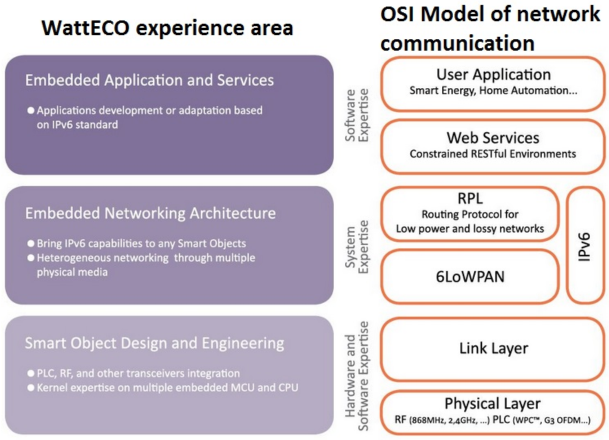

Figure 2.10: WattECO skill area[4]

When using 6LoWPAN, it is possible to expand the reach of IP even further, building low bit rate IP networks over wireless links. From IP backbone, it is possible reach almost any infrastructure location without wires, and therefore share all sensor information for a fraction of the installation costs of non IP wired solutions. All applications can share one single IP network eliminating redundancies. These IP networks can combine multiple physical transmission technologies and even develop over time, without any impact on connected devices.

[4]http://www.watteco.com/

2.4.2 e-SENSE

e-SENSE [21] is a solution to enable Context Capture for Ambient Intelligence by providing an heterogeneous wireless sensor network, particularly for mobile and wireless systems beyond third generation, based on this last communication technology, enables multi-sensory and personal mobile services and applications, as well as assisting mobile communications through sensor information.

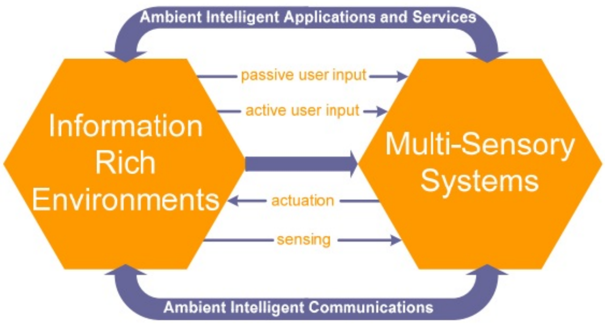

Figure 2.11: Block diagram e-Sense vision [21]

To achieve the project objectives, a consortium of 23 partners has been assembled: 7 industrial partners, 2 small medium enterprises, 4 research institutes and 10 academic institutions. The e-SENSE project is split into the following technology oriented research areas:

- **Energy and Bandwidth Efficient:** wireless sensor communications
- **Scalable and Reconfigurable Transport of Data:** air-interfaces up to the transport layer in a cross functional optimised approach
- **Distributed Processing Middleware:** preparation of data to be presented to upper layers, management of resources and pre-processing of data

e-SENSE topology and interaction of the wireless sensor network depends on the application case, it can be a body sensor network, an object sensor network or and environment sensor network.

2.4.3 Sensei

SENSEI [22] project is an Integrated Project in the European Union (EU)'s Seventh Framework Programme, in the Information and Communication Technologies (ICT) project. This existing project is a leading smart grid solution that provides utilities to enable greater operational efficiencies and improve asset reliability. Sensei includes hardware, software and services especially designed to manage and control networks using intelligent devices communicating between each other, and being remotely monitored. According to Sensei the world is divided in real and digital world. The real world consists of the physical environment that is full with sensors, actuators and processing elements organized in wireless sensor and actuator networks, in order to monitor and interact with the physical entities that we are interested in: people, places and objects. The digital world is divides in three areas:

- Resource: which are representations of sensors and actuators
- Entities of Interest: which are representations of people, places and things
- Resource Users: which represent the physical people or application software that intends to interact with Resources and Entities of Interest

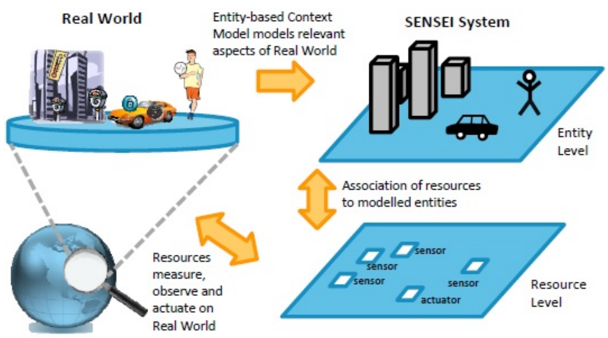

Figure 2.12: Sensei Abstraction levels[5]

[5]https://lh3.googleusercontent.com/-qNubba8XvAc/UP4-hgOwgwI/AAAAAAAAAng/_-es6-6iHrA/s640/SENSEI+abstraction+levels.jpg

2.4.4 Webdust

WebDust [23] is a software platform for managing multiple, heterogeneous (both in terms of software and hardware), geographically disparate sensor networks using a peer-to-peer substrate in order to unify multiple sensor networks. In simple words Webdust organizes, monitors and controls wireless sensor networks, allowing to create customised applications with an easy implementation effort and easy to administrate. WSN is a defined number of distributed autonomous sensors that monitor the physical world and send their data through a network. The WSN is composed of nodes, from just a few to thousands connected wireless, several fundamental problems emerge, such as:

- Hardware and software heterogeneity
- Intermittent connectivity
- Application scaling capability
- Simplification of the development and deployment cycle

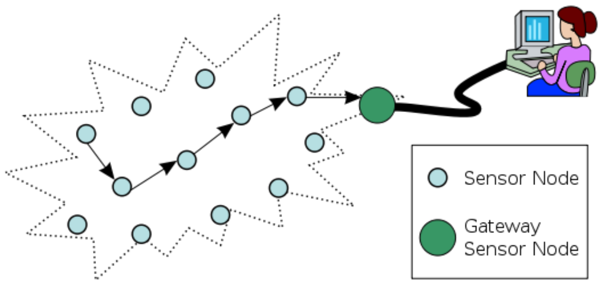

Figure 2.13: Multi-hop wireless sensor network architecture[6]

Looking at Webdust approach to the architecture of the system, we consider global networks as peer-to-peer systems where human users, intelligent agents, and computers interact with wireless sensor networks, ambient intelligence environments and smart spaces, for example. The global network is distinguished in three sub-domains:

- **Peer-to-peer network:** peers are applications on computers, communicating via internet or other networks

[6]http://upload.wikimedia.org/wikipedia/commons/thumb/2/21/WSN.svg/400px-WSN.svg.png

- **Nano-peers:** wireless sensor that communicate via a wireless infrastructure

- **Gateway peers:** this peers are considered to be nodes that have access to the wireless sensor network allowing interaction between peer-to-peer and nano-peer.

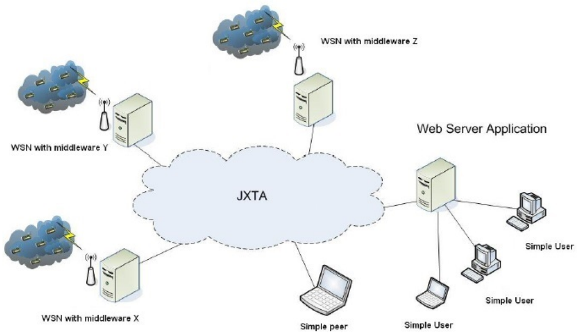

Figure 2.14: Overview of the p2p architecture[7]

2.4.5 BITalino

BITalino [24] is a easy to use scalable platform to analyse real-time bio-signals acquired with a wireless transmission. The concept of a modular device, which allows the user to *mount* and *unmount* the components of the device according to the necessities of the application are in the interest of the dissertation. The device is composed by a micro controller and a group of sensors to monitor physiological data, such as electromyography and electrodermal activity, among others. This technology was designed for user to disassemble the main starting kit and allow him to dispose the independent sensors where is necessary. BITalino comes with a software, Phyton-powered and web-based, capable of direct interaction with an acquisition device. The micro controller block responsible for the convertions of analogic signals from the sensors to a digital format, is a based on AVR 8-bit the clock speed of the system is sourced by an 8MHz external crystal. The BITalino platform includes a low drop voltage regulator (3.3V) powered by a single Lithium Ion Polymer battery with a nominal voltage of 3.7V and 400mA.

[7]Based on: http://www.ourgrid.org/images/OurGrid2.jpg

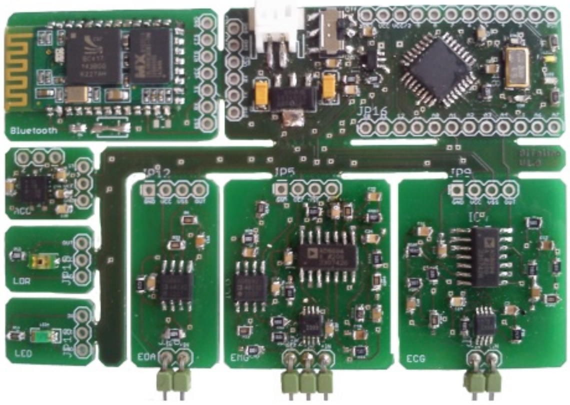

Figure 2.15: The BITalino platform [24]

The physical composition of this platform presents this characteristics: sampling rate is configurable to 1, 10, 100 or 1000Hz, 4 10-bit input analogue ports plus 2 6-bit input ones, 4 input plus 4 output digital ports, data link is composed by a Class II Bluetooth v2.0 with a range up to 10m. The device comes with a serial of sensors that allow physiological recognition together with an accelerometer and an ambient light sensor, the only actuator of the device is a led.

2.4.6 JeeNodes

The JeeNode [25] [26] is a small size wireless board with an Atmel 8-bit Reduced Instruction Set Computer (RISC) microprocessor, is compatible with the Arduino platform and can be programmed with the Arduino IDE. This device is designed for a variety of physical computing tasks, from humidity, measuring and reporting temperature, to tracking and controlling energy consumption. For example the JeeNode v6 has a 6-pin FTDI-compatible serial I/O port, used for power, re-flashing, and communication, the power regulator (3.3V) accepts 3.5V to 13V as external power source. Has a 8-pin combined Power / Serial / I2C. The ATmega328P microcontroller by Atmel, runs with a 16 MHz ceramic resonator, is provided with SPI/ISP interface with 2 general-purpose: I/O lines and wireless radio frequency module that works in 433, 868, or 915 MHz channels. The power consumption of this device varies between $4uA$ to $35mA$, weights $12g$ and measures $85.9x21.1x9.9mm$.

Figure 2.16: JeeNode v6 hardware[8]

[8]Based: on http://jeelabs.net/projects/hardware/wiki//jn#JN6

Chapter 3

ETSI M2M

3.1 ETSI: Organization, History and Objectives

ETSI stands for European Telecommunications Standards Institute [27] which is a leading standardization organization for high quality and innovative Information and Communication Technology standards full filling the various global and European market needs. Founded in 1988, this company, is a non-profit and independent European recognized enterprise whose objective is to create standards, within Europe, to be applied in technological fields such as telecommunication and broadcasting, for example.

3.2 M2M Standard

3.2.1 Mission

The Technical Committee for M2M communication objective is to create open standards for M2M communications to foster the design of a future network of objects and services, so that already existing M2M businesses based on vertical applications can be turned into interoperable M2M services and networked businesses. Vertical proprietary applications shall be substituted by a horizontal architecture, since these applications share common infrastructure, environments, and network elements. An M2M system composed by software/hardware interfaces, protocols, for example, shall ensure the interoperability of all system elements. The Technical Committee's work is based on the general guideline of utilizing existing standardized systems and elements, evaluating them according to M2M requirements filling any gaps as necessary by either enhancing existing standards or by producing supplemental ones.

3.2.2 Framework approach

As already mentioned earlier, to achieve interoperability among M2M, ETSI Technical Committee for M2M communication had to start defining some key features to achieve their goal, some of these characteristic have already been mentioned in the Machine-to-Machine section 2.3.3.

Standardization

One step to achieve interoperability is through standardization, in fact ensuring communication in a standardized ambient is one of the requirements to achieve uniformity. Creating basic primitive functions to be used among all machines is a good start; defining the way that the data is going to be stored and will travel through the network is another important decision to take. These crucial decisions are the ones ETSI TC has to make, to allow standardization.

Vertical to Horizontal

Since it is necessary to provide a system where the high level services are independent from the infrastructure at the base, transforming a vertical solutions into horizontal solutions is an objective. The purpose is to achieve interoperability through a horizontal standardized system, allowing M2M applications to share infrastructures, networks, environments and low-level devices. Designing a structure capable of the above statements, both in standardization and horizontal solution, it is possible to deliver a functional system where applications can be created and modified with, reaching flexibility and management never achieved before.

Security

To achieve interoperability through standardization of an horizontal system, it is important not to minimize security even if the information has low value, due to the variety of applications. Security has to be implemented from the core and applied to every single application, so that it is possible to create instability or differences, that could lead to system unconformity.

Scalability

It is crucial to understand the meaning of scalability, it can be claimed that a system is scalable if the increasing number of devices in the network, does not lead to an infrastructure significant modification. ETSI TC attacks this feature by dividing the network into two main domains, device/gateway and network; increasing the number of devices will only affect the gateway domain. It may seem a paradox that the number of devices does not affect the network domain, since increasing devices should increase the quantity of information flowing, but containing this problem with the right set of frequency and flowing of information in the network.

3.3 M2M Architecture

3.3.1 High-level

The High level architecture for M2M [3] includes a Device and Gateway Domain and a Network domain; in this section an analysis and presentation of the architecture that ETSI TC will be performed, which is mentioned in ETSI TS 102 690 V1.1.1 document.

In Figure 3.1 is represented high level architecture, it can be clearly seen the division between the two domains. The device/gateway domain permits data to travel from the device to the gateway, the device collects data from the *real world* sending it subsequently to the gateway. Since this dissertation focuses in the lower layer, device/gateway domain, an analysis of this elements will be performed with some additional attention. A gateway is a component that runs M2M Application(s) using available M2M Service Capabilities. The Gateway acts as a proxy between M2M Devices and the Network Domain. The M2M Gateway can provide service to other devices connected to it that are hidden from the Network Domain. A device runs M2M application(s) using available M2M Service Capabilities. M2M Devices connect to Network Domain in two manners:

[1] http://forge.fi-ware.org/plugins/mediawiki/wiki/fiware/images/thumb/3/3a/ETSIM2M-HighLevelArchitecture.jpg/500px-ETSIM2M-HighLevelArchitecture.jpg

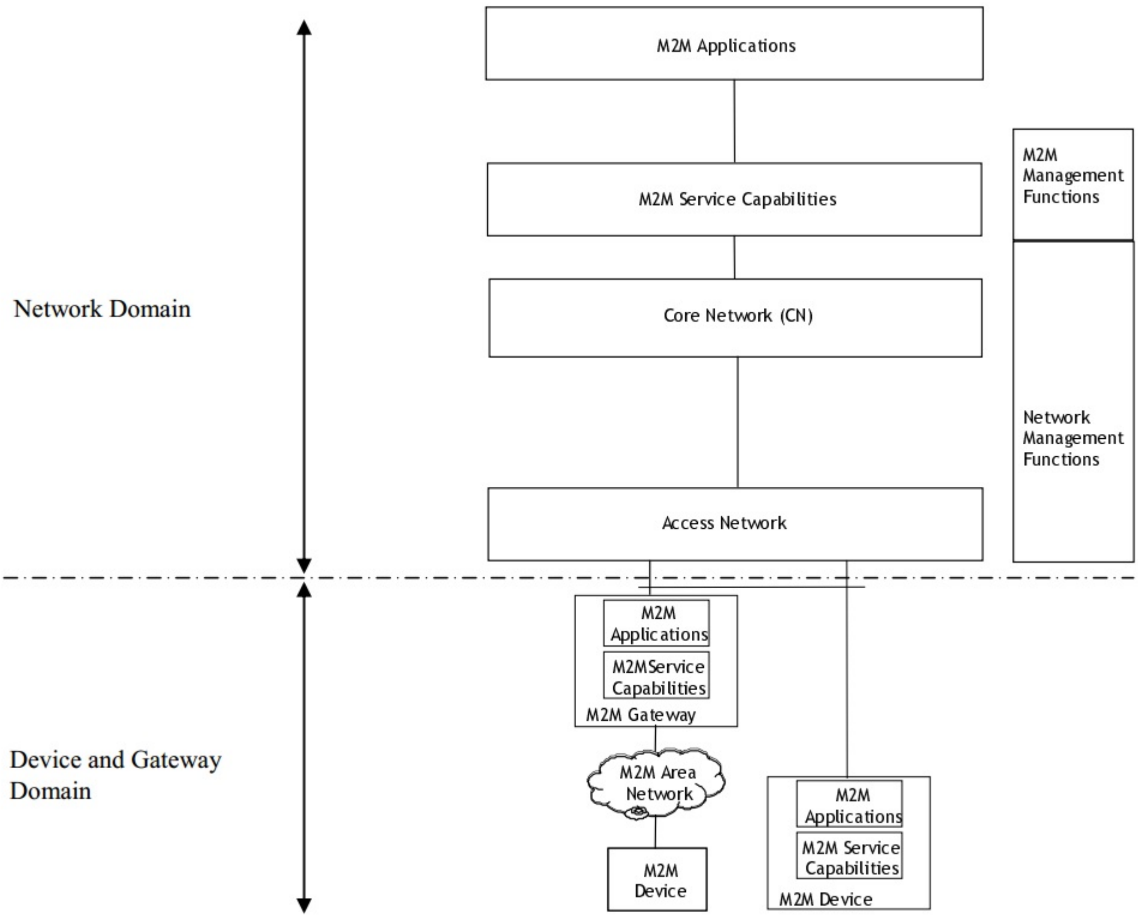

Figure 3.1: High level architecture for M2M, by ETSI[1]

- **Direct Connectivity:** M2M Devices connect to the Network Domain using Access network. The M2M device performs the procedures necessary, being: registration, authentication, authorization, management and provisioning with the Network Domain. The M2M device can also, in case of necessity, provide service to other devices connected to it that are hidden from the Network Domain.

- **Gateway as a Network Proxy:** The M2M Device connects to the Network Domain by M2M Gateway. M2M Devices connect to the M2M Gateway via M2M Area Network. The M2M Gateway acts as a proxy for the Network Domain towards the M2M Devices that are connected to it. Procedures that are *proxied* can include: authentication, authorization, management, and provisioning.

Is important to approach Area Networks, since they appear in the above statements. This feature provides connectivity between M2M devices and M2M gateways,M2M Area Networks include Personal Area Network technologies such as: Institute of Electrical and Electronics Engineers (IEEE) 802.15.1, Zigbee and Bluetooth for example.

3.3.2 Gateway Architecture

Now that an idea on the utility of the gateway was presented, it is possible to explore deeply this component and analyse the blocks and features that compose it, thus the way they interact between each other and the rest of the network above. The capabilities available in the gateway are known as Gateway Service Capabilities Layer (GSCL), the ones available in the devices are called DSCL. Figure 3.2 shows the architecture of the gateway, illustrating the internal components and the reference points of the system. The different components are going to be approached and explained later.

In Figure3.2 it is possible to identify the mId and dIa, these reference points utility is to expose the capabilities of other entities. M2M Service Capabilities Layer provides functions that are to be exposed on the reference points,as already mentioned. Thus M2M Service Capabilities can use Core Network functionalities through a set of exposed interfaces, additionally M2M Service Capabilities can interface to one or several Core Networks.

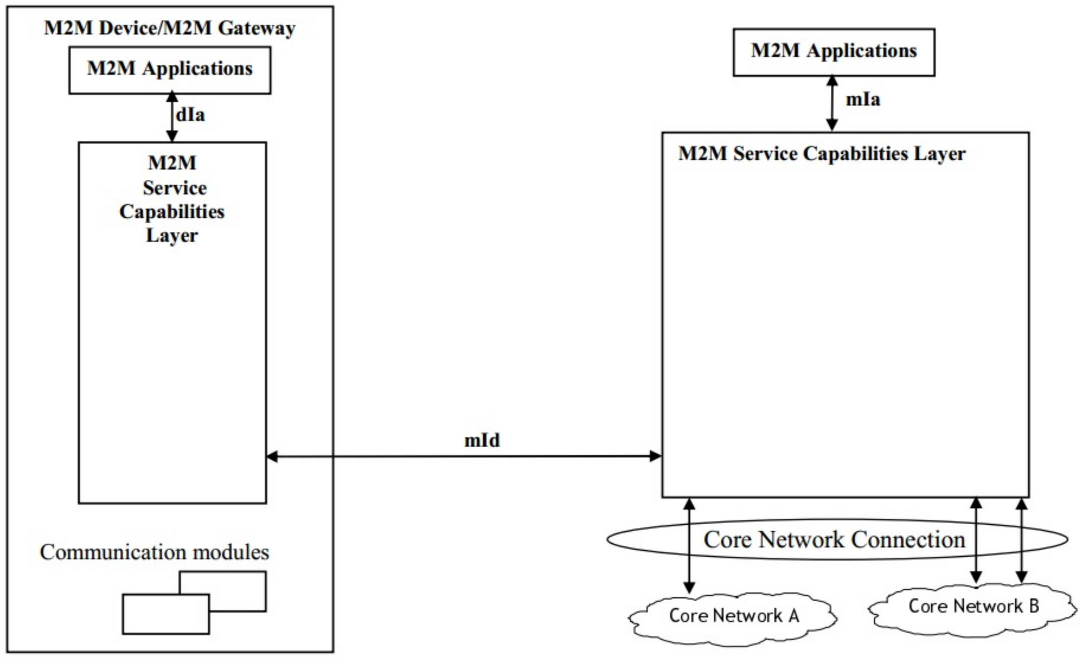

Figure 3.2: GSCL functional architecture, by ETSI [3]

3.3.2.1 Reference point: mId

Communication between the gateway and the network domain is made through this interface, by allowing the Network Application to submit to M2M Service Capabilities in the Network Domain. The mId reference point, include functions such as: registration of a DSCL or a GSCL to the Network Service Capabilities Layer (NSCL); it is able after obtaining the proper authorization to request to read or write information on the DSCL,GSCL or on the NSCL. This entity can request device manager actions, subscription and notification to a specific event, is also able to demand the creation, deletion and listing of a group and provide security.

3.3.2.2 Reference point: dIa

This feature allows the Device and the Gateway Application to access the GSCL. As already clarified ETSI M2M Standard allows two implementations, one including DSCL and another excluding DSCL. The first case, Device Application accesses the GSCL using dIa reference point, depriving of DSCL, and using the gateway as proxy. ETSI recommends some communication protocols such as CoAP and HTTP, both explained in subsection 1.3.5, to allow communication between devices, gateways and network domain. The dIa reference point, between Gateway Application (DA)/Gateway Access (GA) and DSCL/GSCL, include several functions like registration of DA/GA to GSCL and the registration of DA to DSCL, it is also able to request, with a valid authorization, information in the NSCL, GSCL, or DSCL. Like the mId this entity can request the creation, deletion and listing of group(s).

3.3.2.3 Gateway Application (GA)

Gateway applications are a kind of M2M applications which are specifically deployed on top of an M2M gateway. These applications make use of M2M GSCL to filter, process, or apply other specific operations on the underlying data. Afterwards, results are sent upwards in the hierarchy towards the NSCL (conforming to the application's output requirements).

3.3.2.4 GSCL

In this section the focus if given to the entity that provides the capabilities to store and manage data, these capabilities can be accessible to external devices or network domains through the reference points, already explained earlier. The GSCL interacts with the other layers by means of capabilities that are available which are going to be explained.

Gateway Application Enablement (GAE)

The GAE exposes functionalities implemented in GSCL via the single reference point dIa allowing the GA and the DA to register to the GSCL. This capability can provide intra M2M Area Network transmission of a message sent by any application to any other application (GA or DA) connected to the same M2M Gateway. Authorization and authentication to access a specific set of capabilities can be provided to the GA or he DA. Finally it can check if a specific request on dIa is valid before routing it to other M2M Service Capabilities. If a request is not valid an error is reported to the M2M Application (GA or DA)

Gateway Generic Communication (GGC)

This capability provides transport session establishment and tear down along with security key negotiation. If such a transport session establishment is performed in a secure way, then the security key negotiation uses keys as provided by GSEC. Encryption/integrity protection on data exchanged with the NSCL, the Key material for encryption and integrity protection is derived upon secure session establishment, provided by this entity. The GGC ensures secure delivery of application data from/to the Gateway to the NSCL; when security is required the GGC can report transmission errors. This capability has the power to handle name resolution for requests originated within an M2M Area Network and deliver messages in accordance with the Service Class. Relays messages received from GSCL to the Network Generic Communication (NGC) capability in the NSCL and messages received from the NGC Service Capability in the NSCL towards GSCL. The GGC is responsible for the storage and handling of application-specific keys and reports errors triggered by: identifier of the recipient not existing, recipient does not support a requested class, for example by providing secure transport functions such as encryption and integrity protection. More specifically, upon successful M2M Gateway authentication and key agreement, GGC obtains from GSEC application-specific keys for M2M applications associated with the M2M Gateway that are authorized to access NSCL.

Gateway Reachability, Addressing and Repository (GRAR)

The GRAR Provides a mapping between the name of a M2M Device or a group of M2M Devices and a set of information: routable network addresses of the M2M Device, reachability status of an M2M Device and Scheduling information, if available, about the reachability of the M2M Device. It manages subscriptions and notifications pertaining to events and it also allows to create, delete and list a group of M2M Devices. This capability stores GA and DA related registration information and NSCL information, this information can be made available on request or based on subscriptions, subject to access rights and permissions.

Gateway Communication Selection (GCS)

The GCS network selection, based on policies, when the NSCL can be reached through several networks or several bearers and alternative Network or Communication Service selection after a communication failure using a first.

Gateway Remote Entity Management (GREM)

Through this capability the network domain can execute commands in specific devices, in other words the gateway acts like a remote proxy for a defined group of devices. A set of functionalities can be accessed and executed through this capability, thus the gateway acts like a remote management client.

Gateway SECurity (GSEC)

This capability supports M2M Service Bootstrap together with key hierarchy realisation for authentication and authorization initiating mutual authentication and key agreement. If supported by the M2M Gateway, can report the integrity validation status to the NSEC and react to post validation actions triggered by NSEC. The GSEC is responsible for the storage and handling of M2M Connection Keys.

Gateway History and Data Retention (GHDR) optional capability

The GHDR archives relevant information pertaining to messages exchanged over the reference point and also internally to the GSCL based on policies. Can interact with the other capabilities residing in the GSCL to: identify if and which data requires to be retained and report to NHDR history data stored information in the GHDR.

Gateway Transaction Management (GTM) optional capability

The GTM propagates the individual operation requests aggregates the results of the individual operations and commits the transaction when all individual operations have completed successfully, also can trigger a roll-back if any individual operation fails.

Gateway Interworking Proxy (GIP) optional capability

Provides interworking between non ETSI compliant devices and the GSCL; depending on the nature of existing non ETSI compliant M2M deployments, there might be other ways to provide interworking; we also know that full interworking is not possible in some cases, depending on the device/gateway characteristics.

Gateway Compensation Brokerage (GCB) optional Capability

This capability submits compensation tokens to requesting Customers, also verifies the validity of compensation tokens, bills the Customer of compensation tokens for the amount spent. Finally is able to redeem service providers for tokens acquired as compensation for services provided to customers.

3.4 Projects examples

3.4.1 Open MTC

Basing their technology on ETSI TC M2M specifications, this platform was organized into two main services, GSCL and NSCL. This implementation was designed to be applied as an horizontal solution, but is able to support vertical applications. The GSCL supports a variety of M2M area network technology and communication protocols; the NSCL is a cloud-based platform that brings together and saves data from a number of devices acting as an abstraction layer by providing Application Programming Interface (API), through these API it is possible to develop applications to exploit the M2M capabilities.

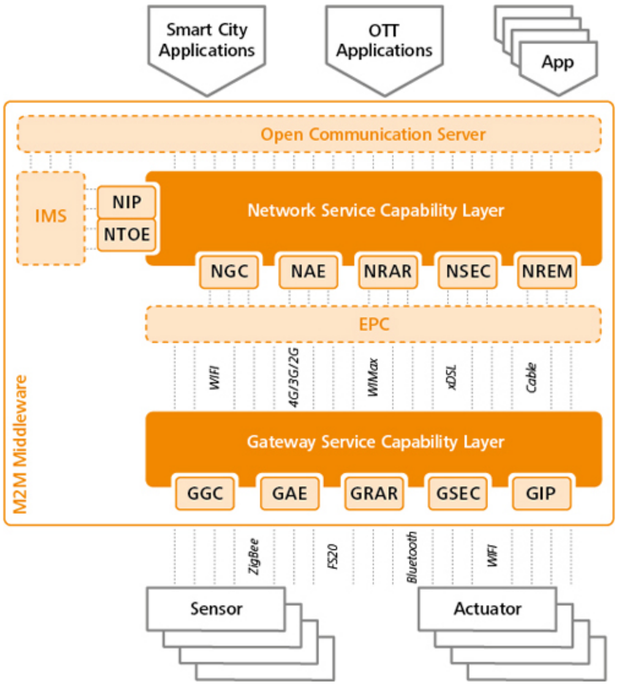

Figure 3.3: Open MTC platform architecture

The first release of the platform was composed by the DSCL, GSCL and NSCL, which have been designed following the early releases of ETSI TS 102.921 and ETSI TS 102.690. This release focuses essentially on the communication aspect of a M2M middleware, enabling:

- **Generic Communication:** provides efficient interaction between different M2M middleware components.

- **Application Enablement:** for communication between sensors and actuators and business front ends; and between application servers and other service platforms on the network side.

- **Reachability, Addressing and Repository:** for lessen the reachability of a large number of connected devices.

- **Interworking Proxy:** for the connection with off-the-shelf machine devices as well as with current operator service platforms.

Looking at Figure 3.3 it is possible to conclude that the gateway, to access the network, has to connect to the Evolved Packet Core (EPC). The IP Multimedia Subsystem (IMS) integrates with M2M middleware is possible through Network Interworking Protocol (NIP). Fraunhofer designed his own architecture for EPC and IMS, and called them respectively openEPC and openIMS.

3.4.2 Cocoon

[28]The project started in 2011 and has been one of the most successful ETSI M2M open source implementations. Cocoon vision is to enable mass market M2M deployments with millions of gateways accessing hundreds of applications, taking advantage of restful API of ETSI M2M to provide a development framework for automation applications which is totally independent of the underlying hardware. Cocoon has an embedded Java environment implementation, which as mentioned use native capabilities like restful interfaces, data containers, data logging, subscribe/notify. The latest version of Object Network Gateway has all interfaces (mId, dIa) have been aligned with release one of ETSI TS 102 921, the protocol driver resource syntax has been aligned with ETSI TR 102 966; all protocol drivers have been redesigned to use the dIa interface over CoAP/UDP; besides ZigBee, 6LoWPAN is available.

Chapter 4

M2M enabled sensor node

The previous chapters presented the technology and allowed to understand and locate the device in the system architecture, it is now possible to examine more deeply the node itself. As it was presented in similar technologies in section 2.4.6, there is the necessity to collect information and send it via wireless to a *listener*, it can be immediately deduced that sensors and a radio frequency module are needed, together with a power source. Figure 4.1 exposes a first approach of the node, through this chapter Figure 4.1 is going to be explored and each component discriminated.

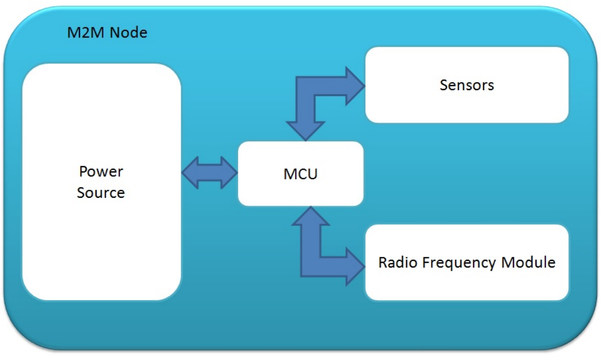

Figure 4.1: Machine-to-Machine node architecture

Follows a more detailed exploration through the requirements and design of the equipment developed.

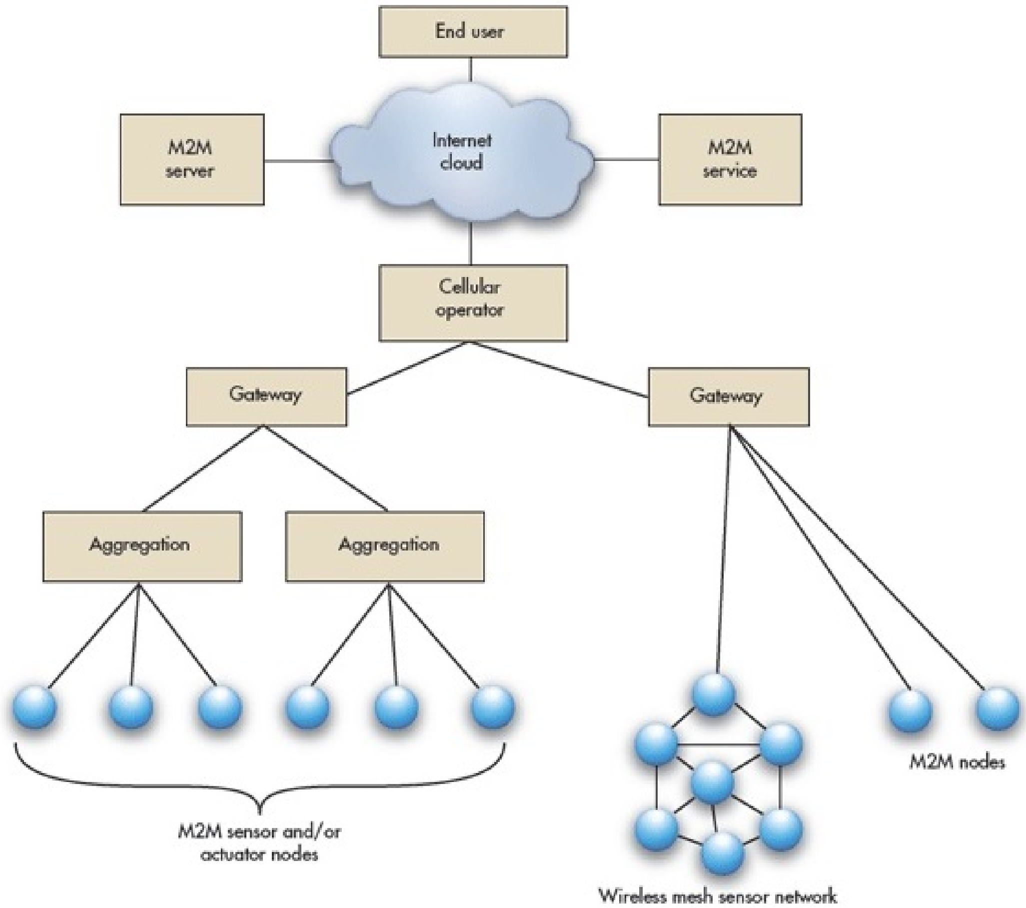

Figure 4.2: Machine-to-Machine basic architecture

4.1 Basic Sensor Node Architecture

The goal using this technology is to sense the environment and communicate the gathered information from the field through wireless links. The network consist of sensor nodes, gateway and a central collecting unit. Since the M2M node connects to a wireless network as shown in Figure 4.2 resuming the general specifications of a wireless sensor network is necessary. The network characteristics are the interface data rate, power output, range, operating frequency and receiver sensitivity. Figure 4.1 is a simple draft, defining potential limitations and requirements that could be faced it is possible to expand the draft obtaining a more detailed diagram. Figure 4.3 shows a more detailed architecture of the node, in the diagram it is possible to distinguish the type of sensors to used, hiding the battery connections since it is not crucial reveal them at this point.

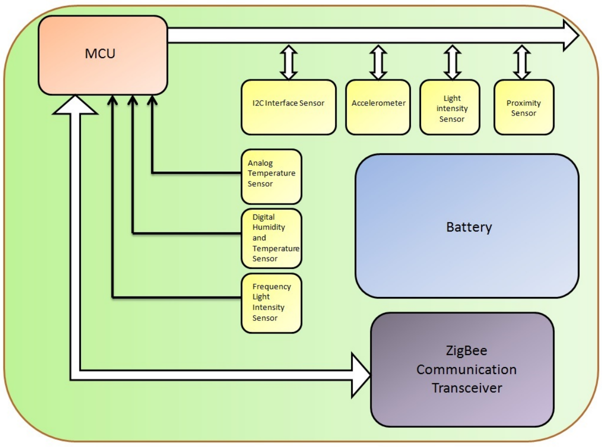

Figure 4.3: Machine-to-Machine detailed sketch

From Figure 4.3 it is possible to conclude what kind of environmental phenomena the device is expected to read: temperature, humidity, acceleration, proximity between others. It is expected from the device to be able at any moment to expand the range of sensors and perform other physical interpretations of the surroundings, so that scenarios like the ones presented in the next section are possible.

4.2 Application Examples

Is now necessary to understand possible applications of a device similar to the one presented in this document. These applications are simple and achievable since the node was designed for a fast deployment and calibration. The crucial factor is that for these two applications it is possible to use the same device saving a considerable amount of human and economic resources.

4.2.1 Agricultural application

[29] Agriculture has been a constant revolution since 1700, bringing collateral effects such as: forcing soil compaction and water scarcity, which leads to landscape shaping and

reducing quality of the environment. The use of heavy machinery in agriculture induces soil compaction, which reduces its capacity to store and conduct water, makes it less permeable for plant roots and increases the risk of soil loss by water erosion, also, water scarcity is becoming one of the biggest problems humankind is facing. It is necessary to modernize the agriculture in Europe, due a huge budget agricultural sector was encouraged to evolve improving irrigation systems and minimizing pollutants. Technology is obviously a tool that can be used to reach the EU's goals in fact WSNs can be used to monitor the physical environment that influence agriculture, parameters like: temperature, humidity, soil temperature/humidity, weather station and leaf wetness, for example, can be monitored to minimize time and money as well as maximize results. The developed technology in this document could be applied in agricultural fields common in Portugal, such as vineyards, the device could be used to monitor environmental parameters such as: ambient temperature, humidity, precipitation, soil moisture, wind, ultraviolet radiation, solar radiation or leaf wetness. Bringing a real time monitoring capable of predicting a plague in the next days or even hours, together with a monitoring of the grape growth from the very beginning to the wine production. Applications that could be implemented on the developed nodes, include capabilities able of getting data from the environmental sensors to create statistical prediction models, controlling the irrigation system and air that condition the agricultural fields. It is also important to notice that the nodes could be powered by lithium battery which are rechargeable by a solar panel or by windmills, making them autonomous. To prevent battery consumption the device can be configured to hold the standby status, basically sleep mode is on most of the time, wake up time is programmable. When the device wakes up reads from the sensors, implements the wireless communication and goes again to sleep mode. The system could complete itself adding pc and smart phone applications to help the user to manage irrigation system and air conditioning machines. This subsection presented a real agricultural application of the M2M technology, using nodes designed in this dissertation.

4.2.2 Smart City Concept Application

The smart city concept [30], as the future of urbanization, has gained huge investments and social capital, due to the increasing urban areas. Since 75% of European Union population chooses cities to live in, this led to a necessary increasing attention over this matter by policy makers. Together with the necessary environmental sustainability it seemed necessary to introduce M2M technology to create a dynamic, interactive and adaptive ecosystem. Taking into account environmental issues such as air pollution and noise, a device such as the one developed in this dissertation could be applied, expanding the system with the proper sensors. Air pollution harms health and nature, it has become crucial to reduce as much as possible nitrogen dioxide and particulate matter emissions, the dangers are local, regional and international. The idea of a constant monitoring has been used to control air standards and can be used to warn authorities when the level of pollutants exceeds predefined levels. Environmental noise can effect humans in physiological and psychological ways, interfering with sleep, rest, study and communication. Monitoring and regulating noise can offer a better quality of life to citizens. This scenario also includes a parking solution for drivers in the city The presented device is able to communicate using 2.4 GHz Zigbee radios using DigiMesh [31] protocols, it is possible to reach 40Km line of sight conditions. As already mentioned in subsection 4.2.1 low power consumption is always critical in these scenarios, the device developed fulfils such requirement, since it was designed with that purpose. Considering pre-calibrated values of the listed parameters, the system reacts when a parameter reaches a critical value and alarm the central node of the occurred, expanding the device to delivery the exact position of the episode is possible through GPS, which could be added to the node.

4.3 Requirements

Sensor networks and their nodes can be applied to very specific scenarios that can have different requirements and limitations, although the main design objectives are mutual. Considering a general scenario a number of demands can be suggested:

- **Adaptability:** Network protocols designed for sensor networks should be adaptive to the circumstance of the grid, in fact a node may fail or even join another structure which could result in a network adjustment of density or topology.

- **Reliability:** Error control and correction mechanisms, that ensure a correct deliver of data over a noisy environment, should be provided by the network protocols.

- **Scalability:** A network protocol designed for a grid of sensors should always be scalable and never take into account the number of devices in the structure, thus the quantity can be in the order of tens or even thousands.

- **Power Consumption:** Having the awareness that, in a sensor network, nodes are powered by battery and it could be very complicated or impossible to recharge the power cluster, energy consumption is crucial and expenditure should be kept always to the minimum.

- **Node Cost:** A sensor node can be deployed in harsh places and in large numbers, we must assume the idea that the node can't be reused for future applications, by knowing all this facts reducing the cost of a single device is imperative since it will influence directly the budget of the project and the right choices can trim the cost of the network.

- **Node Size:** A node with a reduced size is always easier to deploy and can also reduce cost and power consumption.

- **Security:** In a sensor network unauthorized access and malicious attacks must be prevented by security mechanisms, so that data exchange can be sheltered.

- **Misconduct conscience:** A sensor node can be subject to failure or malfunction due to harsh deployment or unattended operations. Thus, sensor nodes should be self-testing, self-calibrating and self-recovering.

- **Channel usage:** Since the bandwidth in a sensor network can be limited, communication protocols should spare resources and improve channel utilization.

- **Auto-set:** When deployed one or more nodes, it is expected that devices are able to organize themselves, subsequently reconfigure their connectivity according to network topology and node failures.

This application focuses on node size, low power consumption, low cost, scalability and adaptability.

Chapter 5

Device Design

5.1 Micro controllers

The processor core of the node is composed by a micro controller, this unit is the *brain* of the device and disposes of memory and programmable input/output peripherals. When comes to choosing this component it is necessary to define the desired features, being them: low power consumption, peripherals, RAM and flash memory and small size. From the typical brands of WSN micro controllers: PIC [32], MSP [33], EFM [34] and Atmega [35]; this last one was taken into account, since it was the more suitable for the node purpose and satisfied the outlined features. The following table compares characteristics of the chosen brand of MCU.

MCU	Power supply	n° pins	n° I/O	FLASH(KB)	Freq(MHz)	type	USB	I2C	UART
ATmega 328p-PU	1.8 - 5.5V	28	23	32	20	dip	no	yes	yes
ATmega645p	2.7 - 5.5V	64	54	64	16	dip	no	yes	yes
Atmega32U4	2.7 - 5.5V	44	26	32	16	smd	yes	yes	yes
ATmega6450P	1.8 - 5.5V	100	69	64	20	smd	n0	yes	yes

Table 5.1: Proposed MCU comparison

5.1.1 Sensor Interfaces

During the development of the project, communication interfaces were used to achieve delivery of data between sensor and observer, and vice versa. Protocols like I2C and SPI were encountered, analogue to digital converters to interpret signals coming from analogue sensors are also found in this application.

I2C - Inter IC: Simplicity and flexibility are key characteristics that make this bus attractive to many applications, most significant features include: two wires (clock and data wires), no strict baud rate requirements, master/slave relationships, devices are software-addressable by a unique address in addition this interface also provides arbitration and collision detection. [36]

SPI - Serial Peripheral Interface: Serial Peripheral Interface [37] is an interface bus used to send data between micro controllers and small peripherals like sensors. It uses separate clock and data lines, along with a select line to choose the device you wish to talk to if needed. This interface is faster than asynchronous serial and supports multiple slaves, but requires more signal lines than other communications methods, communications must be well-defined in advance, master must control all data exchange, and requires separate SS lines to each slave (can be problematic if numerous slaves are needed). In the project only one of the sensors had an SPI interface.

Analogue to Digital Converter: An Analogue to Digital Converter (ADC) [38] is feature that converts an analogue voltage on a pin to a digital number. We can use electronics to interface to the real world around us by converting to the digital world. One of the most common technique uses the analogue voltage to charge up an internal capacitor and then measure the time it takes to discharge across an internal resistor. The micro controller monitors the number of clock cycles that pass before the capacitor is discharged, the number of cycles is the number returned by the ADC. This application uses a 10 bit ADC that permits to divide the voltage range in 1023 levels, each level is an interpretation of the physical world, using the following relation:

$$\frac{Resolution of the ADC}{System Voltage} = \frac{ADC Reading}{Analogue Voltage Measured}$$

Figure 5.1 presents the blocks diagram of the device including the communication interfaces described in this section and used by the MCU to interact with the sensors and the rest of the components of the node.

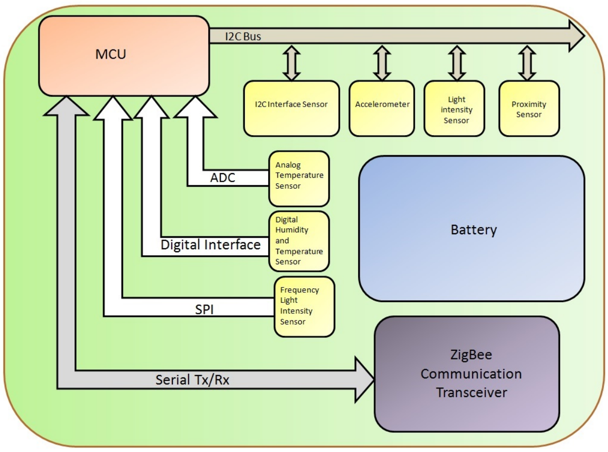

Figure 5.1: Machine-to-Machine sketch including communication interfaces

5.2 Sensors

A sensor is a device that converts physical quantity into a signal that can be read by an observer, like mercury in thermometer is use to check temperature: while measuring temperature mercury expands or contracts, by doing so it is possible to observe through a calibrated glass tube the temperature. In this project the observer, a micro controller, extracts values so that after a necessary format the transceiver can send them to higher levels. From a large variety and quantity of sensors some were chosen to perform this dissertation and many variables were taken into account allowing the more suitable choice, being them: price, power consumption, size and communication interface.

5.2.1 Light Sensors

Light to Frequency Converter - TSL235R: The TSL235R [39] light-to-frequency converter outputs a square wave (50% duty cycle) with frequency directly proportional to light intensity. The device has been temperature compensated for the ultraviolet-to-visible light range of 320 nm to 700 nm and responds over the light range of 320 nm to 1050 nm. The TSL235R can be powered at anywhere between 2.7 and 5.5V, and will typically pull about 2mA of current. It comes in a simple, 3-pin through hole package. No external components are necessary to use the converter, and the digital output allows direct interface to a micro controller.

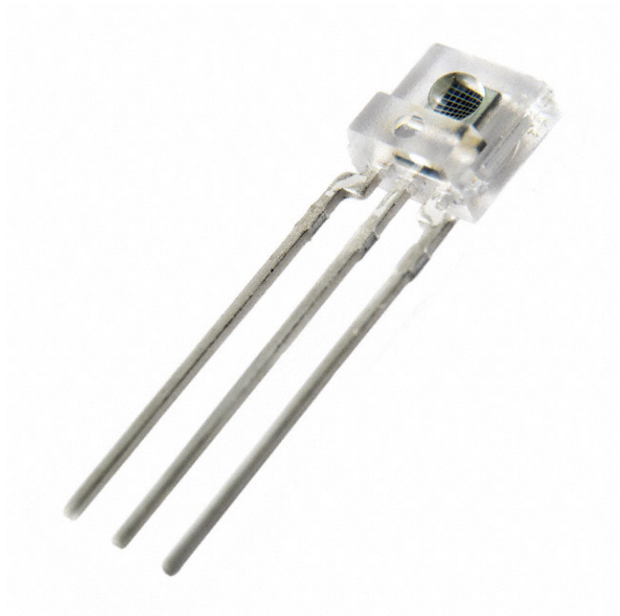

Figure 5.2: TSL235R-LF[1]

This component was chosen for electrical characteristics that allow low consumption, low price and fast configurable interface. Since other light sensors were chosen, the TSL235R stands out from the others because of the communication interface, the component uses SPI instead of I2C.

5.2.2 Temperature Sensors

Humidity and Temperature Sensor - AM2302: Also known by DHT-22 [40], is a low cost humidity and temperature sensor with a single wire digital interface. The sensor is calibrated and does not require extra components. The AM2302 can be powered at anywhere between 3.3 and 6V, and will typically pull about 40-50uA standby current and 1-1.5mA measuring current. Has ranges of humidity from 0 to 100% with ±2% accuracy and -40 to 80 degrees Celsius temperature range with ±0.5 degrees accuracy.

[1]http://media.digikey.com/Photos/AMS-Taos%20USA%20Photos/TSL235R-LF.jpg

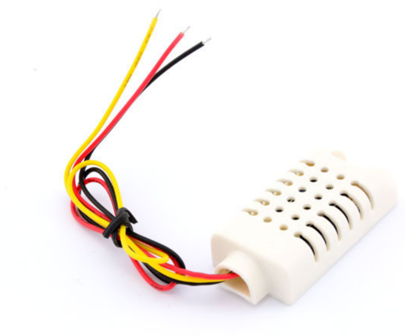

Figure 5.3: AM2302[2]

The AM2302 was chosen due to low power consumption, low price and it is an easy component to be installed and programmed. **Temperature Sensor - TMP36:** The TMP36 [41] is a low voltage, precision centigrade temperature sensor that provides a voltage output linearly proportional to Celsius temperature, does not require any external calibration to provide typical accuracies of ±1°C at ±25°C and ±2°C over the -40°C to + 125°C temperature range. The 30-pin device needs a ground and 2.7 to 5.5V to work, voltage on the Vout pin can be read for further conversion with a scale factor of $10mV/^{o}$C.

[2]http://www.kiwi-electronics.eu/image/cache/data/products/componenten/sensors/COMP-AM2302-800x533.jpg

Figure 5.4: TMP36[3]

The TMP36 was chosen for low power consumption characteristics and a very low price. This component has a very fast reading time and uses an Analogue to Digital Converter (ADC) as interface with the MCU. Analogue sensors on board gives an alternative to digital ones, in applications where it is necessary to use a vast number of temperature sensors the TMP36 is a valuable alternative to the more expensive digital ones.

5.2.3 Accelerometer

Triple Axis Accelerometer - MMA8452Q: The MMA8452Q [42] is a electromechanical low-power device, three-axis, capacitive Micro-Electro-Mechanical Systems (MEMS) accelerometer with 12 bits of resolution capable of measuring acceleration forces. The MMA8452Q has user selectable full scales of $\pm 2g/\pm 4g/\pm 8g$ with high pass filtered data as well as non filtered data available real-time. The device can be configured to generate inertial wake-up interrupt signals from any combination of the configurable embedded functions allowing the MMA8452Q to monitor events and remain in a low power mode

[3]http://media.cdn-libelium.com/catalog/product/cache/1/image/9df78eab33525d08d6e5fb8d27136e95/1/0/10988-01.jpg

during periods of inactivity. The device communicates with the MCU via I2C.

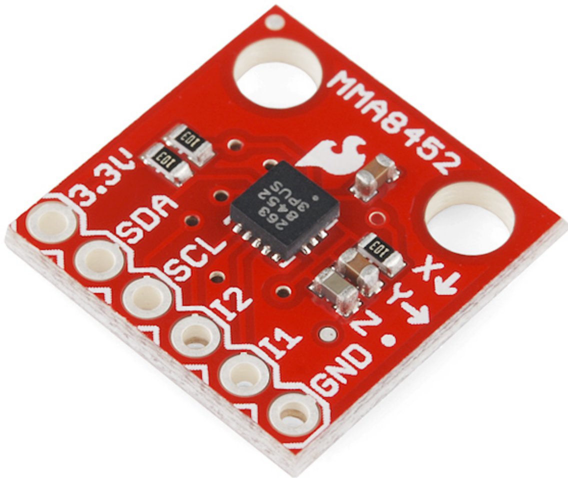

Figure 5.5: MMA8452Q with breakout[4]

The Triple Axis Accelerometer was chosen to give the node a physical interpretation of the world, which can be used for several applications, for example monitor if an object was touched or moved. The component provides embedded functions with flexible user programmable options, configurable to two interrupt pins. The functions allow for power savings saving the MCU from continuously polling data.

5.2.4 Proximity Sensor

Infrared Light and Proximity - VCNL4000: The VCNL4000[43] can detect proximity to an object using infra-red within a range of 20cm. Proximity data as well as ambient light level data is collected over an I2C interface. The VCNL4000 can be powered at 3.3V, a separated power source is available for the the infra red emitter that can be powered between 2.5 and 5V.

[4]https://oceancontrols.com.au/images/D/ACC-025_1.jpg

Figure 5.6: VCNL4000 with breakout[5]

The component was chosen due to low power consumption, I2C interface and the ability to turn on and off one of the two sensors on board, allowing for further power savings.

5.3 Communication Transceivers

A transceiver is a communication device that receives and transmits data, in simple words provides a link between two ends. Researching the appropriate radio frequency device it was possible to discover many WSN specialized brands/transceivers, many of these did not satisfy pre-established requirements, indeed only technologies working in lower regimes than the GHz bands were considered, since range and power consumption are imperial in the process of selection. Higher frequencies present two major adverse circumstances: signal attenuation and power consumption. Alternative ways of communication exist such as WiMax [44],LTE [45], Wifi [46], Bluetooth [47] and GPRS [48] or even Ethernet [49]. Thus, this dissertation considers ZigBee [15] as communication technology, deepened in the following section. The following table compares examples of similar exist-

[5]http://media.cdn-libelium.com/catalog/product/cache/1/image/9df78eab33525d08d6e5fb8d27136e95/v/c/vcnl4000-infrared-emitter-breakout.jpg

Device	Frequency(MHz)	Transmission Consumption(mA)	Receiving Consumption(mA)	Data Rate(Kbps)	Transmission Power(dBm)
AT86RF212	863-870	9.2	17.5@5dBm,700MHz	20-1000	10
CC1101	300-928	14.2	29.2@10dBm,433MHz	0.6-600	12
Si4455	283-960	10	18@10dBm,868MHz	0.5-500	13
MRF48XA	433/868/915	11	22@7dBm,433MHz	0.6-256	7
ZMD4410	868/915	29	28@10dBm,868Mz	0-40	0

Table 5.2: Proposed transmission module comparison

ing technologies: ZigBee [16] is a dominant brand when it comes to WSN, it was chosen due to power savings and network organization capability.

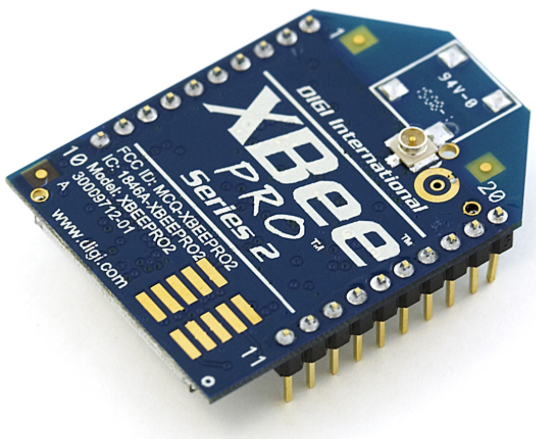

Figure 5.7: ZigBee PRO Series 2[6]

This brand was chosen over the others due to the capabilities it presents, such as: wireless range up to 70m indoors and 400m outdoors, highly scalable solutions for thousands of devices, supports multiple network topologies, operation band according to IEEE 802.15.4.

[6]https://cdn.sparkfun.com//assets/parts/4/8/9/4/10420-03-L.jpg

5.3.1 Communication Protocol

To satisfy ETSI requirements and to to strengthen data exchange a communication protocol was applied. Communications protocols [50] cover authentication, error detection and correction, and flagging. They can also describe the syntax, semantics, and synchronization of analogue and digital communications. Communications protocols can be implemented by hardware and/or software. The chosen communication protocol was CoAP, it is easy to implement and effective to use. This protocol international model works based on internet packets, to request a device's data: sends a GET to a device and then expect back a packet with the requested data, it could be a sensor value for example. It is also possible to push a packet of data to a device: as a POST. The payload chosen was JSON format, its composed by an ordered list of values. In most programming languages, this is realized as an array, vector, list, or sequence. The JSON is characterised for *lightweight*, and it was easy to implement and interpret between the layers evolved.

5.4 Power Supply

The battery supply of the node is an important component, in fact the more it lasts the more profitable the device becomes. Recalling the requirements of the node, for example size and weight, available clusters in the market had to be analysed and required a selection based on the specifications.

Battery	Nominal Capacity	Voltage(V)	Maximum Continuous Discharge(mA)	Dimensions(mm)	Weight(g)
GSP58360	2.0Ah@20mA , to 3V	3.7	2000	60.5x53.3x6.0	18.5
U10004	1.5Ah@10mA, to 1.5V	3.3	125	48x45x4.4	15

Table 5.3: Proposed battery comparison

From the table is possible to conclude that the Ultralife U10004 lithium non-rechargeable battery [51] presents the suitable characteristics for the dissertation purpose, voltage ranges are acceptable and the current maximum discharge is reasonable when taking into account the energy consumption 6.3 of the device.

5.5 Remote Access and Control

It is important to control and monitor nodes of the WSN, these two functionalities are dependent from each other, in fact if a node would need to act on a given circumstance it must have information about what is happening for a proper action to be taken.

5.5.1 Monitoring and Control

Real time monitoring is crucial to receive information about the scenario where the sensor nodes are applied, this data could be use to diagnose the system. There are no restrictions about the sensors that are used, neither the retrieved information from them. To control nodes in the system it is important to establish a centralized control, since it yields a more flexible system. Depending on the application, it is possible to configure several values of the device to attune functionalities [52].

5.6 Conclusion

In this last chapter information about the structure of a basic sensor node was acquired, limitations and requirements were establish. Sensors and communication supports were analysed, compared and chosen as components for the final device of this dissertation.

Chapter 6

Case study

The final chapter presents the design of the project which was developed, implemented and tested. The data collected from the different test is going to be analysed and compared in the different test scenarios, so that it becomes possible to draw reliable conclusions. Following the performed analysis of similar devices in previous sections, it was possible to develop a node capable of facing different scenarios meeting the defined requirements.

6.1 Device prototype

In this section the node is going to be presented and physically described to provide a corporeal idea of what was attained by the selection of all the components that fulfilled the requirements established. Figure 6.1 shows the preliminary device still on breadboard that was used to perform the necessary tests and allows to discern some of the components on board.

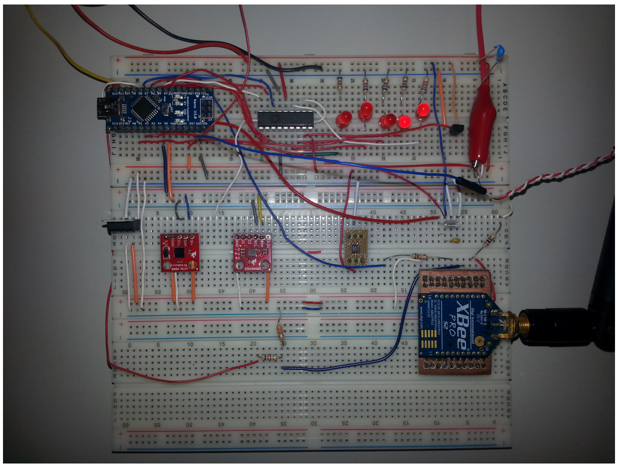

Figure 6.1: Device prototype picture

The board presents a variety of components, all of them already specified in the precious chapter, is possible to highlight a diversity of sensors, both analogue and digital, able to measure temperature, humidity, light intensity, proximity and acceleration. On board is also available an external ADC, a 8:1 multiplexer, a 3V3 voltage and finally the communication module with the appropriate antenna. In the above Figure 6.1 passive components can be seen, which allow the correct function of the electrical circuit. The components of the device are controlled and data is processed by an ATmega 328p-PU, which was installed, configured and programmed to respond to the gateway applications.

6.2 Implementation of the system

The components were chosen to allow the system to be flexible. Starting from the MCU the system was thought for a low power consumption being able to perform functions both at 8MHz and 16 MHz, being the first one the least grim on the battery. The MCU has been configured to be able to process different communication interfaces so that connecting a new sensor can be an easy step, including hardware and software process. The privileged interface in this architecture is an I2C bus with a software scanning support that delivers the sensor address, henceforth adding the correct class to the library the new sensor can be used in minutes. Other interfaces are available such as SPI, ADC and even if not a communication interface a number of digital ports are free for further expansion. The SPI interface is available and can be used even if the interruption ports are limited, in this project one of the sensors, more specifically the frequency light intensity sensor, connects to the MCU through SPI. The multiple power lines can allow a more efficient usage of the battery, in fact there are sensors and other components that able to work at different voltages, a lower voltage can be chosen to decrease energy consumption. When possible the sensors were connected to the minimum delivered voltage of the device. Power lines on board are fixed at 5V and 3V3, this voltage lines can be activated and deactivated by the MCU through the multiplexer, *keeping alive* only the peripheral that is going to be used in that moment. The external ADC is located on board to allow a supplementary energy saving since it is possible to deactivate the internal ADC of the MCU, it becomes an option powering the ADC allowing further power savings. The internal ADC of the MCU is available if necessary, but in this dissertation application it was chosen to deactivate the interface, for power saving strategies. The ADC that was chosen also allows MCU port sparing, since this component has an addressable I2C interface, giving the node a generous expansion capacity for analogue device connections. The ZigBee module connects to the MCU through serial interface, both for data receiving and transmission. Thus it was necessary an adaptation since the MCU is powered at 5V and the radio frequency module is fed at 3V3, a simple resistive divider was applied and the correct function of the transceiver was ensured. The system starts up enabling the necessary interfaces and sensors, after this short time event an I2C scan is performed to recognize the addresses existing on the data bus, by retrieving this information the node achieves awareness of the sensors available to perform their functions. The next step executed is the bootstrap between the DSCL and the GSCL, the accomplishment of this task allows the synchronization of the two independent systems, which are ready to exchange information. The device may read from sensors and transmit or reconfigure some of the pre-established tasks, for example recalibrate transmitting periods. After the MCU composes the JSON payload containing the sensor values, the transmission module is solicited to perform the bootstrap with the Gateway, this step is performed via software using the radio frequency component, the data in then transmitted over CoAP from the node to the Gateway.

{"sensor":{"N":"TMP36","T":"A","V":"28.61","U":"C" }}

Figure 6.2: JSON payload from one sensor

The JSON provides information such as the name of the sensor, the type, the value read and the units. From Figure 6.2 is possible to retrieve crucial data, for example the sensor's name is *TMP36*, this device is an analogue sensor and is reading *28.61* in Celsius degrees. Figure 6.2 presents only the information of one sensor, the payload is significantly bigger if all the sensors are questioned. A simple example of how the device operation was presented so that it is now possible to present the performed tests and their correspondent analysis, it is important to mention that the device was only challenged indoors and, an outdoor tests could be sightly different.

6.3 Tests and Data Analysis

To test the device a selection of tests were performed in order to understand the strength and the weaknesses of the project. It seemed necessary to challenge the energy consumptions, packages latency packages and software issues. The power consumption was the main focus of the tests, since software can be optimized during a real application of the device and data packages latency can depend on a number of variables that could not be simulated in laboratory. For the device software and energy consumption to be challenged a series of scenarios were thought:

- Case 1: system on reading from five sensors on a variety of interfaces
- Case 2: system on reading from five sensors on a variety of interfaces and transmitting the read values using CoAP to reach gateway
- Case 3: system reading from one analogue sensor and transmitting to gateway using CoAP
- Case 4: system reading from one digital sensor attached on I2C bus and transmitting to gateway using CoAP

	Current [mA]	Voltage[V]	Power[mW]
Case 1	46	5	230
Case 2	46+19	5	325
Case 3	33+16	5	245
Case 4	31+17	5	240
Case 5	0.02	5	95

Table 6.1: Device consumption values

- Case 5: Sleeping system

After the performing tests the first approximation of energy consumption can be predicted, and a more suitable battery can be selected for the application purpose.

After an analysis of the current draw values, it is possible to justify the u selection of the U10004 battery cell as referred in subsection 5.4, since the maximum continuous discharge of current is able to handle the necessity of the device. The second table presents the consumption of some sensors available in the device, both digital and analogue, the current measurement was performed with an amperemeter, as for the first tests, it seemed important to mention that only consumption during the reading phase of the sensor was considered, since during the rest of the time the sensors are turned off and the current consumption is more significant during this stage.

	Current Consumption[mA]	Voltage[V]	Power[mW]	Type
VCNL Proximity	0.0073	3.3	0.00239	Digital
VCNL Light	51.17	3.3	168.9	Digital
IR Frequency	1.6	5	7.5	Frquency
TMP36	0.04	5	0.2	Analogic
MMA8452	1.8	3.3	5.94	Digital

Table 6.2: Sensors consumption values

Before discussing the duration of the battery and perform theoretical calculations, further data is necessary such as: time spent during readings and transmissions. These test were performed via software with time stamp functions, which were processed and from them it was possible to retrieve the approximate time necessary to perform the given tasks.

	Time 0[ms]	Time 1[ms]	Time 2[ms]	Mean[ms]
Bootstrap	1382	7941	1834	3719
Case 2	2899	2859	2846	2868
Case 3	95	94	92	94
Case 4	410	409	411	410

Table 6.3: Device duty time measurements

Table 6.3 reveals the time needed for the device to return values read by the sensors. For instance to know the time the device takes to perform the bootstrap with the gateway, read five sensors and respond the questioner via radio frequency module the answer can be retrieved from Case 2 of the table. The measurements of the bootstrap row were performed independently from each other Case, this measurements were taken via software with time stamp functions. The measurements of Case 2, 3 and 4 were performed after the bootstrap was concluded, so that to know the total latency it is necessary to add the bootstrap time with Case 2,3 or 4. For example a Case 2 latency:

$$1382 + 2899 = 4281\ ms$$

The bootstrap times can vary, due to software issues and communication synchronization, the values vary from approximately 1 to 8 seconds, nothing significantly can be done to give this necessary process a smaller variance. A simulation of the system with the tested values was performed and presented in the next sections; a section dedicated to discuss the calculations anticipates a section presenting the system performance conclusions.

6.3.1 System performance

After performing the consumption tests and assuming a battery of 1600mA it is possible to stipulate the autonomy the the device. To keep current consumptions low as possible, so that the system can stay as long as possible in the field, it is necessary to turn off several other unnecessary functions and internal devices so that the node can reach an expected current draw of approximately 20uA, this theoretical value that can be achieved in future developments can lead to a profitable usage of the battery, in fact. Since the sensors are only turned on when interrogated the the consumption of current is limited to the reading time, as for the Zigbee module which has a very low consumption during the *sleeping phase* and consumes current only when questioned. In *sleeping condition*, the device should last ideally more than a year; but it is not expected from this type of technology not to perform any physical interpretations with the available sensors and transmit this values to the Gateway to be further processed. With the measured duty periods and the correspondent power consumption is possible to simulate endurance of the battery. Picturing a scenario where only one sensor is needed with two readings every hour and another where five sensor are needed performing readings every 2 hours. Some auxiliary values and calculations were necessary so that it was possible to perform a prediction of time duration of the battery. For example, seconds in one day:

$$24 \times 60 \times 60 = 86400 \ seconds$$

Battery capacity in milliamperes per second instead of the value given in milliamperes per hour:

$$1600 \times 60 \times 60 = 5760000 \ mAs$$

The calculations were performed by establishing the amount of time the device would be working and the amount time the device would be in standby. The time values can be retrieved table 6.3. Together with consumption values retrieved from table 6.1 it is possible to obtain the consumption for the active state in one day for each scenario established. The remaining seconds of the day are considered as standby status, where the device consumption is significantly lower. These seconds are calculated from the subtraction of the active state time from the total seconds in one day, the result allows to obtain the consumption of the standby state in one day. Adding the active state with the standby state consumption gives the total consumption of the device in one day; the total capacity of the battery divided by the calculated consumption value results in the total theoretical days the battery could be able to last. In addition an efficiency coefficient that multiplies the final result was chosen to provide a more realistic result to the calculation predictions.

6.3.1.1 Simulation: five sensors, two readings each hour

The simulations are based on values from the test section, such as: bootstrap time, reading time, current consumption and the total capacity of the battery, for the time values the mean of the readings for each case was used. The title of this section already defines the characteristic of this scenario: five sensors performing two readings per hour and transmitting data to the gateway. The calculations were performed following the methodology explained in the last part of subsection 6.3.1 as Figure 6.3 presents.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  2 readings per hour, 5 sensors                    %
% Tb => bootstrap time                               %
% Tr => reading and transmission time                %
% To => time on                                      %
% Tod=> time on in a day                             %
% Cr => current consumption during readings          %
% Cs => current consumption during sleep             %
% Cb => battery capacity                             %
% Ccdo=> Current consumption per day                 %
% Ccds=> Current consumpion sleep state per day      %
% Ct => total consumption per day                    %
% Tac => total active state day consumption          %
% Tsc => total sleep state day consumption           %
% Nd => number of days battery capacity              %
% Bc => battery capacity                             %
Tb = 3.7;       %   s                                %
Tr = 2.8;       %   s                                %
Cr =  65E-3;    %   mA                               %
Cs = 20E-6;     %   mA                               %
Bc = 1600E-3;   %   AH                               %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
To =  Tb + Tr;
Tod = To*24;

Ccdo = Cr * 2 * 24;
Ccds = [(24 * 60 * 60)-Tod] * Cs;

Ct = (Ccdo + Ccds);
Cb = Bc*3600;

Nd = Cb/Ct;
Nd = Nd*0.85
% Nd =
%
%   1010.6
```

Figure 6.3: Calculation sketch: five sensors, two readings each hour with transmission

From the calculations it was possible to draw graphics which allow a visual interpretation of the device working periods, these graphics were adapted to the document by reducing and increasing scales.

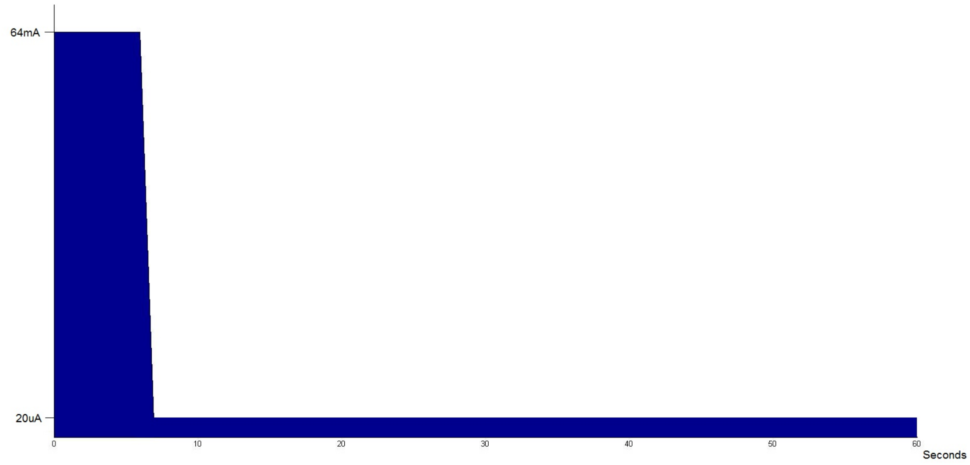

Figure 6.4: Simulation graphic 60 seconds

Figure 6.4 shows the behaviour of the device consumption over one minute, the node operates for approximately 6.5 seconds and then enters the standby status.

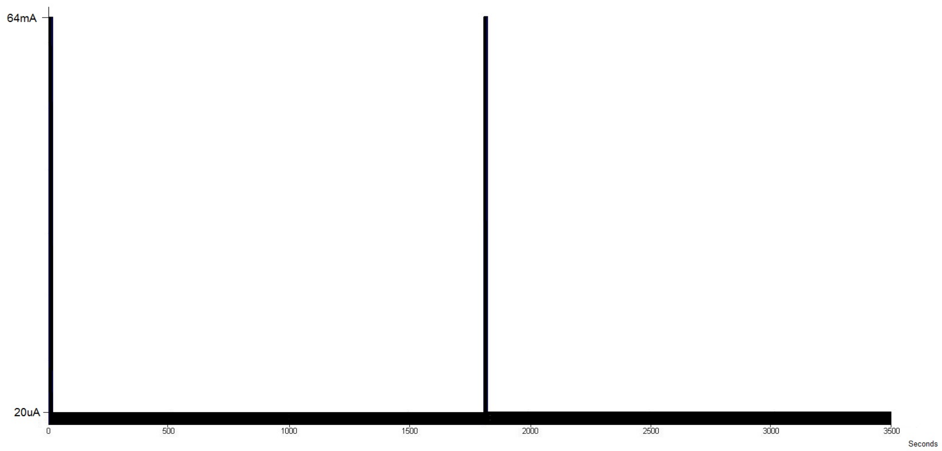

Figure 6.5: Simulation graphic 3600 seconds(1 hour)

Figure 6.5 shows the behaviour of the device consumption over one hour, the node operates for approximately 6.5 seconds as in Figure 6.4 and then enters the standby status, in this diagram is possible to distinguish the two pre-established readings performed by the sensors. The 6.5 graph was not meant to give precise data, but only to give a graphical example of the system duty cycle.

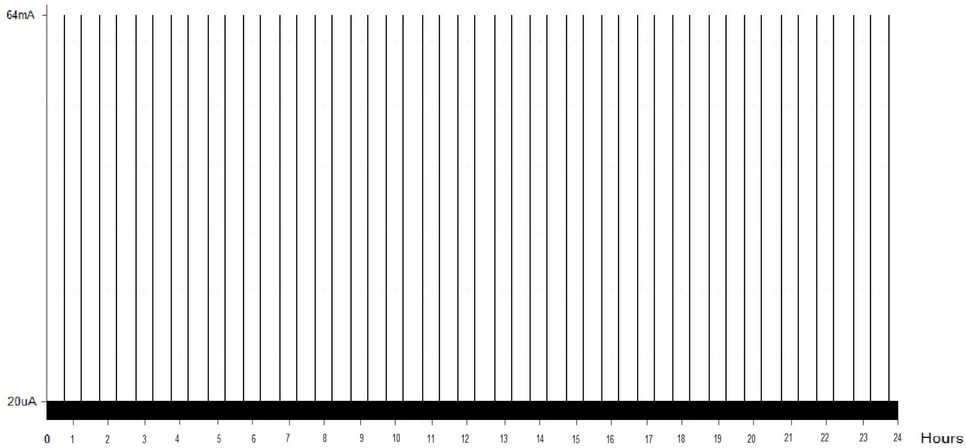

Figure 6.6: Simulation graphic 86400 seconds(24h)

Figure 6.6 Finally Figure 6.6 presents an explicit final overlook over the 24 hours of the device application in the defined scenario. During the 24 hours of a day the device performs the readings, transmits and goes back to standby status. Since the test values were used it is possible to distinguish the consumption value of 64mA when on duty and the 20uA when on standby.

6.3.1.2 Simulation: five sensors, five readings each hour

This simulation, as for the previous one, was calculated based on the values obtained in the test section. The scenario simulated pictures the device performing five readings, digital or analogue, and transmitting via radio frequency module. The calculations were performed following the methodology explained in the last part of subsection 6.3.1.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   5 readings per hour, 5 sensors                 %
% Tb => bootstrap time                              %
% Tr => reading and transmission time               %
% To => time on                                     %
% Tod=> time on in a day                            %
% Cr => current consumption during readings         %
% Cs => current consumption during sleep            %
% Cb => battery capacity                            %
% Ccdo=> Current consumption per day                %
% Ccds=> Current consumpion sleep state per day     %
% Ct => total consumption per day                   %
% Tac => total active state day consumption         %
% Tsc => total sleep state day consumption          %
% Nd => number of days battery capacity             %
% Bc => battery capacity                            %
Tb = 3.7;       %   s                               %
Tr = 2.8;       %   s                               %
Cr = 64E-3;     %   mA                              %
Cs = 20E-6;     %   mA                              %
Bc = 1600E-3;   %   AH                              %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
To =  Tb + Tr;
Tod = To*24;

Ccdo = Cr * 24*5;
Ccds = [(24 * 60 * 60)-Tod] * Cs;

Ct = (Ccdo + Ccds);
Cb = Bc*3600;

Nd = Cb/Ct;
Nd = Nd*0.85
% Nd =
%
%   520.6
```

Figure 6.7: Calculation sketch: five sensors, five readings each hour with transmission

The graphs in this section were maintained with the *Matlab* format, since modifying the figures was not advantageous for the document.

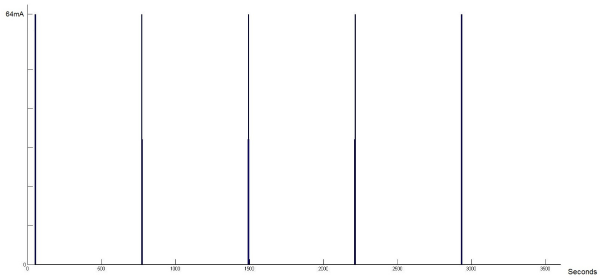

Figure 6.8: Simulation graphic 3600 seconds(1 hour)

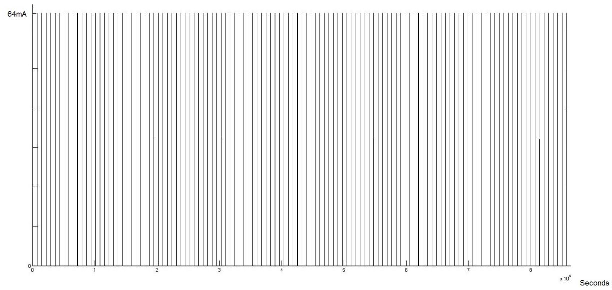

Figure 6.9: Simulation graphic 86400 seconds(24 hours)

Comparing both the simulations of the sections it is possible to conclude that the second simulation in section 6.3.1.2 consumes more battery, in fact comparing Figure 6.9 with Figure 6.6 the frequency *stipes* representing the on state is significantly higher. In the Discussion section a graphical interpretation of the battery consumption is going to be analysed.

6.4 Discussion

The node power consumption compared with other similar devices shows a modest consumption behaviour, in fact it could last more than a year in the field without any battery substitution. Depending on the scenario the energy cell may last more or less time, for example in an appliance where more surrounding readings and more wireless transmissions are needed the battery would be consumed rapidly and may last half of the time; thus a new battery type, with other characteristics is needed. Figure 6.10 and Figure 6.11, present the over time depletion of the battery.

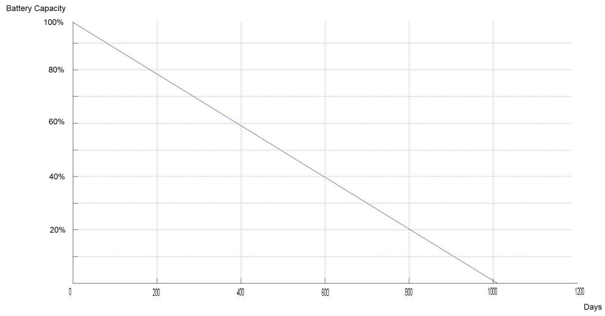

Figure 6.10: Battery Consumption for: five sensors, two readings each per hour

Figure 6.10 is the graphical representation of simulation in section 6.3.1.2.

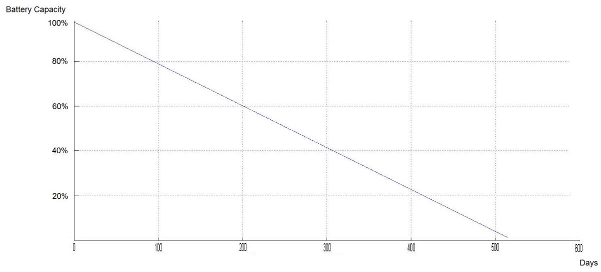

Figure 6.11: Battery Consumption for: five sensors, five readings each per hour

Figure 6.11 is the graphical representation of simulation in section 6.3.1.1.

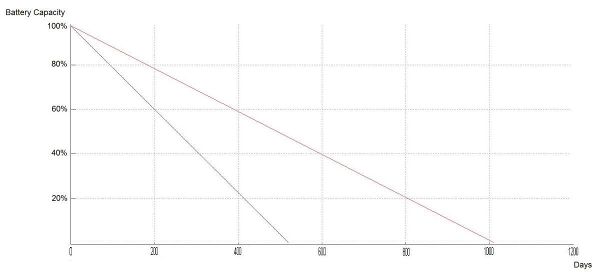

Figure 6.12: Battery Consumption comparison

In Figure 6.12 is possible to compare the two consumption scenarios, concluding that more readings and more data transmissions lead to a faster depletion of the battery stored energy. It is important also to notice that the battery does not present a linear consumption in a real application, since physical aspects like for example temperature influence the behaviour of the energy cell, extending and reducing durability. It is possible to simulate and perform calculations for a high amount of scenarios, thus only these two were chosen. From Table 6.2 further analysis are possible, in fact this table distinguishes the consumptions of each sensor used during the tests, discriminating analogue and digital components and their consumption. For instance in an application where a continuous monitoring of temperature is necessary and precision is not crucial, the analogue TMP36 sensor is the appropriate device to satisfy the requirements in fact the low power consumption and the fast interaction with the MCU give this peripheral advantage over the digital AM2302 available, which consumes more. Thus if it is necessary to read humidity together with temperature this last sensor could be more useful than the TMP36, since as mentioned in section 5.2.2 the AM2302 is able to measure humidity, saving PORTS of the MCU. The software is able to function with all the components mounted or with only a selection of them, adding or removing a class from the library can save the device memory space. The node designed in this dissertation, memory space was not an issue, but looking at further sensor extension it is important to have notion that the device memory is limited.

Chapter 7

Conclusion

The primary objective of this dissertation was to design and create and intelligent device that could be applied on several different fields with different necessities. The design was achieved after the main device components were chosen between various types and with different characteristics. Modernization is a crucial topic when it comes to scenarios where power savings and system optimisation can lead to substantial capital savings. It was possible to achieve a relatively cheap product able to obey to the ETSI standard regulations and able to embrace the IoT world with interoperability characteristics. The WSN node is able to collect data from the surrounding as it was tested, and maintain power consumption to the lower value possible. The CoAP protocol was implemented and tested with the GSCL and the NSCL successfully. On the other hand, it seems necessary check for performance issues, ensuring that the gateway could handle more demanding scenarios, like for example instead of having one single device, have a bigger number with higher data streaming. The architecture defined that the node implementation allows for a sensor expansion or reduction. Some test were performed so that the performance of the device could be stipulated, facing a deployment scenario. Besides the system explanation the document also presents a serial of scenarios and projects, where device like the one developed are used and are *the eyes and ears* of the system. This practical cases are useful to better understand the architecture of this type of systems. Finally the dissertation also provides an overview of the ETSI standard and the vision of this approach.

7.0.1 Future developments

Future developments include design and manufacture of a PCB for the sensor node, the layout should take into account the sensors, the power necessity and size of the node. Following this first step a period of more harsh testing must occur so that all the requirements: performance versus size, cost, power consumption between others can be evaluated and a reliable final product can proceed to an economical evaluation. From the hardware point of view the device could be expanded to a new bundle of sensors or even actuators, depending on the scenario the node is to be applied, keeping the supple characteristic presented to the design that was presented in this dissertation. To achieve robustness, ultra-low power consumption, reliability, adaptability and scalability it's necessary to debug and attune software and all the appropriate protocols. The software running in the device could be optimized for a decreased processing necessity and improve the "sleep"/"awake" abilities of the device, reducing even more the power consumption. The continuous evolving ETSI project may force new requirements on the software, to achieve interoperability it is necessary to keep the device updated. Like any technology existing there is always a margin for improvement leaving the imagination as the limit.

Bibliography

[1] E. Way, "Harvesting the Power of Ubiquitous Sensor Networks," no. March, pp. 57–64, 2009.

[2] ETSI, *Etsi ts 102 689*, 2010, vol. 1.

[3] T. Specification, "Etsi ts 102 690," vol. 1, pp. 1–280, 2011.

[4] ETSI, *Etsi ts 102 921*, 2012, vol. 1.

[5] Julien Mineraud and Sasu Tarkoma, "Internet of Things Magazine," pp. 1–52, 2014.

[6] Internet of Things Initiative, "Internet of things initiative," 2014. [Online]. Available: http://www.iot-i.eu/public

[7] M. Hatton, "The Global M2M Market in 2013," 2013. [Online]. Available: http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:The+Global+M2M+Market+in+2013#2

[8] Z. M. Fadlullah, M. M. Fouda, N. Kato, A. Takeuchi, N. Iwasaki, and Y. Nozaki, "Toward intelligent machine-to-machine communications in smart grid," *IEEE Communications Magazine*, vol. 49, no. 4, pp. 60–65, Apr. 2011. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5741147

[9] "SunSPOTWorld - Home -," 2014. [Online]. Available: http://www.sunspotworld.com/

[10] S. F. Barrett, "Arduino Microcontroller: Processing for Everyone! Second Edition," *Synthesis Lectures on Digital Circuits and Systems*, vol. 7, no. 2, pp. 1–371,

Jun. 2012. [Online]. Available: http://www.morganclaypool.com/doi/abs/10.2200/S00421ED1V01Y201205DCS038

[11] W. Datasheet, "Waspmote," 2012.

[12] ARM, "Platforms | mbed," 2014. [Online]. Available: https://mbed.org/platforms/

[13] O. H. David Boswarthick, Omar Elloumi, *M2M Communications: A Systems Approach*, 2012. [Online]. Available: http://books.google.com/books?hl=it&lr=&id=bVaqAFpH6EgC&pgis=1

[14] P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta, and Y. F. Hu, "Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards," *Computer Communications*, vol. 30, no. 7, pp. 1655–1695, May 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366406004749

[15] P. Kinney, "ZigBee Technology : Wireless Control that Simply Works," no. October, pp. 1–20, 2003.

[16] N. X. P. L. Uk, "ZigBee PRO Stack User Guide," no. June, 2014.

[17] J. Zheng and M. J. Lee, "A Comprehensive Performance Study of IEEE 802.15.4," pp. 1–14.

[18] P. Chang, L. M. Vittie, T. M. Manager, and A. Services, "The Fundamentals of HTTP."

[19] G. Business, B. A. Simonnet, A. Simonnet, T. Capital, O. Hersent, S. G. Paris, E. Berthaud, P. Bertrand, A. Store, and S. Electric, "Smart grid : Actility repense la stratégie de Watteco," pp. 2010–2012, 2012.

[20] E. Lehmann, "6LoWPAN Fundamentals White Paper," no. July, 2012.

[21] W. S. Alexander Gluhak, "A WSN System Architecture to Capture Context Information for beyond 3G Communication Systems," 2007. [Online]. Available: http://ieeexplore.ieee.org/Xplore/guesthome.jsp?signout=success

[22] V. Tsiatsis, A. Gluhak, and T. Bauge, “The SENSEI Real World Internet Architecture.” *Future Internet ...*, pp. 247–256, 2010. [Online]. Available: http://ebooks.iospress.nl/Download/Pdf/30737

[23] I. Chatzigiannakis, C. Koninis, G. Mylonas, U. Colesanti, and A. Vitaletti, “A peer-to-peer framework for globally-available sensor networks and its application in building management,” *...on Sensor Network ...*, 2009. [Online]. Available: http://www.researchgate.net/publication/228972797_A_peer-to-peer_framework_for_globally-available_sensor_networks_and_its_application_in_building_management/file/d912f50eeb495ce168.pdf

[24] J. Guerreiro, R. Martins, and H. Silva, “BITalino-A Multimodal Platform for Physiological Computing.” *ICINCO (1)*, 2013. [Online]. Available: http://www.researchgate.net/publication/259177489_BITalino_A_Multimodal_Platform_for_Physiological_Computing/file/3deec52a225606fb83.pdf

[25] L. Russell, A. Steele, and R. Goubran, “Low-cost, rapid prototyping of IMU and pressure monitoring system using an open source hardware design,” *...Conference (I2MTC), 2012 ...*, 2012. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6229719

[26] “JeeNode - Hardware - JeeLabs . net,” 2014. [Online]. Available: http://jeelabs.net/projects/hardware/wiki/JeeNode

[27] ETSI, “ETSI - European Telecommunications Standards Institute,” 2014. [Online]. Available: http://www.etsi.org/

[28] O. C. Website, “Cocoon overview | Cocoon.” [Online]. Available: http://cocoon.actility.com/documentation/ongv2/overview

[29] “Smart Agriculture project in Galicia to monitor a vineyard with Waspmote | Libelium.” [Online]. Available: http://www.libelium.com/smart_agriculture_vineyard_sensors_waspmote/

[30] R. Tanaka, M. Schmidt, and Y. Takamatsu, “An Energy Awareness Study in a Smart City Lesson Learned,” no. April, pp. 21–24, 2014.

[31] D. International, “XBee & XBee-PRO DigiMesh 2.4 Embedded RF modules - Datasheet.”

[32] “Microchip MCUs | Microchip Technology Inc.” [Online]. Available: http://www.microchip.com/pagehandler/en-us/products/picmicrocontrollers

[33] “Low-power Microcontroller | 16-bit Microcontroller | 16-bit MCU - Overview - MSP - TI.com.” [Online]. Available: http://www.ti.com/lsds/ti/microcontrollers_16-bit_32-bit/msp/overview.page

[34] “Mixed-Signal 8-bit and 32-bit Microcontroller Devices | Silicon Labs.” [Online]. Available: http://www.silabs.com/products/mcu/Pages/default.aspx

[35] “Atmel AVR 8-bit and 32-bit Microcontrollers.” [Online]. Available: http://www.atmel.com/products/microcontrollers/avr/default.aspx

[36] E. D. Transfer, “Tackling I 2 C Development Complexities Using Innovative Tools,” pp. 1–11.

[37] M. Sekhar, “Design and Verification of Serial Peripheral Interface using OVM,” *ijecct.org*, vol. 2, no. 6, pp. 267–269, 2012. [Online]. Available: http://www.ijecct.org/v2n6/(267-269)0206M10.pdf

[38] A. D. C. Selection and P. Literature, “Adc08100,adc12081,adc12181,adc12281.”

[39] F. B. Diagram, A. Options, and T. Functions, “TSL235R LIGHT-TO-FREQUENCY CONVERTER Texas Advanced Optoelectronic Solutions Inc . TSL235R LIGHT-TO-FREQUENCY CONVERTER,” no. September, pp. 1–10, 2007.

[40] L.-t. Stability, “Digital relative humidity & t emperature sensor AM2302 : 1 . Feature & Application Application : 2 . Description : 3 . Technical Specification :,” pp. 1–5.

[41] F. B. Diagram, P. I. N. Configurations, and G. Description, “Low Voltage Temperature Sensors,” 2010.

[42] T. Mmaq, "12-bit / 8-bit Digital Accelerometer," pp. 1–47, 2013.

[43] V. Semiconductors, "Vishay Semiconductors Fully Integrated Proximity and Ambient Light Sensor with Infrared Emitter and I 2 C Interface VCNL4000," no. 1, pp. 1–14.

[44] A. Note, "General information about Wimax," 2006.

[45] T. Lte and G. P. P. Release, "Technology Introduction White Paper," 2009.

[46] "WLAN / Wi-Fi Fundamentals - Rohde & Schwarz." [Online]. Available: http://www.rohde-schwarz.com/en/solutions/wireless-communications/wlan-wifi/fundamentals/wcdma-fundamentals_106601.html

[47] C. Bisdikian and Y. Heights, "IBM Research Report An Overview of the Bluetooth Wireless Technology An Overview of the Bluetooth Wireless Technology," vol. 22109, no. June, 2001.

[48] j. C. Goodman and D. J., "General Packet Radio Service in GSM," *IEEE Communications Magazine*, 1997.

[49] S. Ruping, E. Vonnahme, and J. Jasperneite, *Fieldbus Technology*, D. Dietrich, H. Schweinzer, and P. Neumann, Eds. Vienna: Springer Vienna, Jan. 1999. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-7091-6421-1_46

[50] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, vol. vol.1. IEEE Comput. Soc, 2000, p. 10. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=926982

[51] T. Datasheet, "10004 : Thin Cell," pp. 331–332, 2014.

[52] L. Grieco and M. B. Alaya, "Architecting information centric ETSI-M2M systems," *... Workshops), 2014 IEEE ...*, pp. 211–214, 2014. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6815203