

UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE  
TELECOMUNICACIÓN



FINAL THESIS REPORT

M.S. Degree in Telecommunication Engineering

DEVELOPMENT OF A MULTIPLE RF  
INTERFACED PLATFORM FOR COGNITIVE  
WIRELESS SENSOR NETWORKS

Agustín Tena García

November 2013



# FINAL THESIS REPORT

**Title:** DEVELOPMENT OF A MULTIPLE RF INTERFACED PLATFORM FOR COGNITIVE WIRELESS SENSOR NETWORKS

**Author:** AGUSTÍN TENA GARCÍA

**Advisor:** ELENA ROMERO PERALES

**Academic Advisor:** ALVARO ARAUJO PINTO

**Departament:** DEPARTAMENTO DE INGENIERÍA ELETRÓNICA

## THESIS COMMITTEE

**President:** D. ÁLVARO DE GUZMAN FERNÁNDEZ GONZÁLEZ

**Member:** D. ALVARO ARAUJO PINTO

**Secretary:** D. PEDRO JOSÉ MALAGÓN MARZO

**Assistant:** D. MIGUEL ÁNGEL SÁNCHEZ GARCÍA

**SCORE:**

Madrid, a de de



UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE  
TELECOMUNICACIÓN



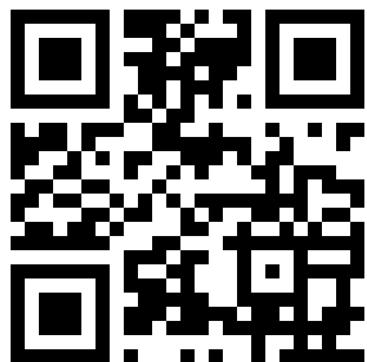
**FINAL THESIS REPORT**

M.S. Degree in Telecommunication Engineering

**DEVELOPMENT OF A MULTIPLE RF  
INTERFACED PLATFORM FOR COGNITIVE  
WIRELESS SENSOR NETWORKS**

Agustín Tena García

November 2013



Online version. <http://goo.gl/mQ3Mez>.

Document formatted with TEXIS v.1.0+.

This document is prepared for duplex printing.

All the source files used to generate this document can be accessed at the public repository:  
<https://github.com/agus-xyz/cNGD-mem>.

# Resumen

Actualmente, las redes de sensores inalámbricas están sometidas a impedimentos que dificultan su desarrollo y despliegue. Un ejemplo de estas limitaciones es la creciente saturación del espectro radioeléctrico en las bandas de frecuencia libres.

Las redes cognitivas, apoyándose en un modelo de comunicación cooperativo, representan un nuevo paradigma que tiene por objetivo un uso más eficiente del espectro y la mejora de las comunicaciones inalámbricas. Las redes de sensores cognitivas aplican propiedades cognitivas en redes de sensores comunes, desarrollando así nuevas estrategias para mitigar la ineficiencia en las comunicaciones que sufren estas redes.

Es importante investigar modelos cognitivos para explorar los beneficios que pueden aportar a nuestras redes de sensores, especialmente limitadas en energía y recursos. Sin embargo, pocas plataformas y con funcionalidades generalmente pobres o demasiado específicas permiten su estudio debido a un estado del arte aún inmaduro. La mayoría de las investigaciones se llevan a cabo sobre simuladores, los cuales muestran resultados parciales o incompletos.

Este documento presenta el diseño y la implementación una plataforma versátil que integra propiedades cognitivas en las redes de sensores. Se han combinado módulos hardware y software en un instrumento que ayude a la investigación de las redes de sensores cognitivas. La plataforma permite comunicación sobre tres bandas distintas de radiofrecuencia, siendo la única plataforma para redes de sensores que lo permite, y el hardware se ajusta a exigencias de tamaño, coste y energía. Además, su diseño modular y escalable es ampliamente adaptable a cualquier aplicación para redes de sensores.

**KEY WORDS:** *redes cognitivas, radio cognitiva, redes de sensores inalámbricas, plataforma, banco de pruebas, nodo, dispositivo, redes de sensores inalámbricas cognitivas.*



# Abstract

Nowadays, WSNs (Wireless Sensor Networks) are subject to development and deployment constraints, such as the increasing RF spectrum saturation at the unlicensed bands.

CNs (Cognitive Networks), leaning on a cooperative communication model, represent a new paradigm aimed at improving spectrum efficiency and wireless communications. CWSNs (Cognitive Wireless Sensor Networks) compound cognitive properties into common WSNs, thereby developing new strategies to mitigate the inefficiency in communications.

It is important to investigate cognitive models to explore their benefit over our WAHSNs, specially constrained in energy and resources. However, few platforms allow their study due to their early research stage, and they still possess scarce or specific features. Investigations take place mainly over simulators, which provide partial and incomplete results.

This dissertation presents the development of a versatile platform that brings together cognitive properties into WSNs. Hardware and software models are combined to create an instrument used to investigate CWSNs. The hardware fits WSN requirements in terms of size, cost, and energy. It allows communication over three different RF bands, and it thus the only cognitive platform for WSNs with this capability. In addition, its modular and scalable design is widely adaptable to almost any WAHSN application.

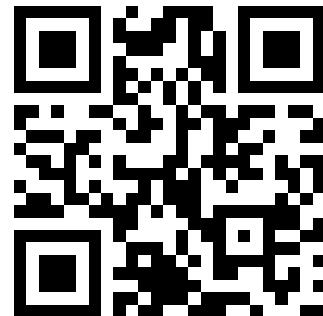
**KEY WORDS:** *cognitive networks, cognitive radio, wireless sensor network, platform, testbed, cognitive wireless sensor network, node, device.*



*A mis abuelos, padres e hijos*



# Acknowledgements



*<http://tiny.cc/oymm5w>*



# Index

<b>Resumen</b>	<b>VII</b>
<b>Abstract</b>	<b>IX</b>
<b>Acknowledgements</b>	<b>XIII</b>
<b>1. Introduction</b>	<b>1</b>
1.1. Background . . . . .	1
1.2. Goals . . . . .	3
1.3. Project Organization . . . . .	4
1.4. Outline . . . . .	5
<b>2. Cognitive Wireless Sensor Networks: State-of-the-art</b>	<b>7</b>
2.1. Cognitive Radio . . . . .	7
2.2. Cognitive Networks and Cognitive Radio Networks . . . . .	10
2.2.1. A simple example . . . . .	11
2.3. Wireless Sensor Networks . . . . .	12
2.4. Cognitive Wireless Sensor Networks . . . . .	15
2.4.1. Current Implementations . . . . .	17
2.4.1.1. Software platforms - simulators . . . . .	18
2.4.1.2. Hardware platforms . . . . .	20
2.4.1.3. Standards . . . . .	29
2.5. Contribution . . . . .	29
<b>3. Review Study</b>	<b>31</b>
3.1. Cognitive Wireless Sensor Network Node Model . . . . .	31
3.2. First Cognitive Device review . . . . .	33
3.3. Firmware review . . . . .	35
3.4. Cognitive Radio Module review . . . . .	35
3.5. System Requirements . . . . .	36
3.6. Technology . . . . .	37
3.6.1. Microcontroller . . . . .	37
3.6.2. Radio Interfaces . . . . .	38
3.6.3. Serial Communication . . . . .	39
3.6.4. Power Supply System . . . . .	40
3.6.5. Timing . . . . .	40
3.6.6. Expansion Capabilities . . . . .	40

3.7. Conclusions . . . . .	41
<b>4. Design Study</b>	<b>43</b>
4.1. Global Hardware Description . . . . .	43
4.2. Transceivers - $\mu$ Trans 434/868 . . . . .	45
4.2.1. Description . . . . .	46
4.2.2. Schematics . . . . .	49
4.3. Main Board - cognitiveNextGenerationDevice . . . . .	49
4.3.1. Description . . . . .	49
4.3.2. Schematics . . . . .	50
4.3.3. Microcontroller . . . . .	50
4.3.3.1. Pin-out . . . . .	53
4.3.4. Radio Interfaces . . . . .	55
4.3.4.1. Software Controlled Power Supply . . . . .	58
4.3.5. Power Supply System . . . . .	59
4.3.5.1. Battery . . . . .	61
4.3.6. Timing . . . . .	61
4.3.7. Expansion Headers . . . . .	63
4.3.7.1. Pin-out . . . . .	64
4.3.8. PGE . . . . .	65
4.3.9. Others . . . . .	66
4.4. Serial Communication Board - rs232SHIELD . . . . .	66
4.4.1. Description . . . . .	66
4.4.2. Schematics . . . . .	67
4.5. Batteries charger - chargerSHIELD . . . . .	67
4.5.1. Description . . . . .	67
4.5.2. Schematics . . . . .	69
4.5.3. Simulation . . . . .	69
4.6. Tools . . . . .	70
4.6.1. Altium Designer . . . . .	70
4.7. Components Library: cngd lib . . . . .	71
<b>5. Implementation Study</b>	<b>73</b>
5.1. Introduction . . . . .	73
5.1.1. Tools . . . . .	74
5.2. Transceivers - $\mu$ Trans 434/868 . . . . .	74
5.2.1. Description . . . . .	75
5.2.2. Layout . . . . .	75
5.2.3. Prototype mounting and testing . . . . .	76
5.2.4. Final Result . . . . .	76
5.3. Main Board - cognitiveNextGenerationDevice . . . . .	77
5.3.1. Description . . . . .	77
5.3.2. Layout . . . . .	79
5.3.3. Prototype mounting and testing . . . . .	79
5.3.4. Final result . . . . .	79
5.4. Serial Communication Board - rs232SHIELD . . . . .	80

5.4.1. Description . . . . .	80
5.4.2. Layout . . . . .	81
5.4.3. Prototype mounting and testing . . . . .	81
5.4.4. Final results . . . . .	81
5.5. Battery charger - chargerSHIELD . . . . .	81
5.5.1. Description . . . . .	81
5.5.2. Layout . . . . .	83
5.5.3. Prototype mounting and testing . . . . .	83
5.5.4. Final results . . . . .	83
5.6. Conclusions . . . . .	84
<b>6. Testings and Results</b>	<b>85</b>
6.1. Test Model . . . . .	85
6.2. Energy management test . . . . .	85
6.3. Computing test . . . . .	87
6.4. Spectrum sensing test . . . . .	87
6.5. Radio interfaces agility test . . . . .	88
6.6. Effective rate test . . . . .	88
6.7. chargerSHIELD performance test . . . . .	90
<b>7. Software</b>	<b>93</b>
7.1. Tools . . . . .	93
7.1.1. MPLAB X . . . . .	93
7.1.2. Programmer: ICD 3 . . . . .	93
7.2. Firmware and Hardware Abstraction Layer . . . . .	94
7.2.1. Platform options . . . . .	94
7.2.2. USB tracing . . . . .	95
7.2.3. Radio Interfaces Power Control Functions . . . . .	96
7.2.4. Headers and other new definitions . . . . .	96
7.3. Demo Application Layer . . . . .	96
<b>8. Conclusions</b>	<b>99</b>
8.1. Further Studies . . . . .	101
<b>9. Estimated Costs</b>	<b>103</b>
9.1. Physical implementation cost . . . . .	103
9.2. Overhead and industrial profit . . . . .	104
9.3. Fees for project management and writing . . . . .	104
9.4. Total budget . . . . .	104
<b>I Appendixes</b>	<b>105</b>
<b>A. Schematics</b>	<b>107</b>
<b>B. Layouts &amp; Components</b>	<b>121</b>

<b>References</b>	<b>137</b>
<b>List of Acronyms</b>	<b>145</b>

# List of Figures

1.1.	World different regions defined by the ITU . . . . .	2
1.2.	Electromagnetic frequency spectrum . . . . .	2
2.1.	A snapshot of the spectrum utilization up to 6 GHz in an urban area at BWRC . . . . .	8
2.2.	Main functions of the PHY, MAC, and network layers in a CR . . . . .	9
2.3.	Simple relay network for a wireless network . . . . .	12
2.4.	Possible WSN implementations . . . . .	13
2.5.	WSN node general model . . . . .	13
2.6.	Common WSN protocols . . . . .	15
2.7.	Comparison of the number of hops per packet in a CWSN simulated scenario . . . . .	16
2.8.	NS2 and NS3 simulators logo . . . . .	18
2.9.	OMNeT++ simulator logo . . . . .	18
2.10.	Castalia simulator logo . . . . .	19
2.11.	NetSim simulator logo . . . . .	20
2.12.	SENDORA simulator logo . . . . .	20
2.13.	Picture of the expanded First Cognitive Device . . . . .	21
2.14.	Pictures of VESNA and SNE-ISMTV modules . . . . .	22
2.15.	Libelium WaspMote . . . . .	23
2.16.	Libelium Meshlium . . . . .	24
2.17.	CREW Project scheme overview . . . . .	24
2.18.	One floor deployment of TWIST test-bed . . . . .	25
2.19.	LOG-a-TEC nodes clusters deployment . . . . .	26
2.20.	Node locations of the w-iLab.t test-bed . . . . .	27
2.21.	Deployment of VT-CORNET test-bed . . . . .	27
2.22.	CorteXlab test-bed logo . . . . .	28
2.23.	CorteXlab test-bed deployment . . . . .	29
3.1.	CWSN node model . . . . .	32
3.2.	Picture of the FCD, developed at LSI in 2011 . . . . .	33
3.3.	Architecture model of the FCD . . . . .	34
3.4.	CRmodule architecture . . . . .	36
3.5.	MRF49XA PICtail <sup>TM</sup> Daughter Board . . . . .	39
3.6.	Main wireless networking options comparison . . . . .	39
4.1.	Global hardware modules of the platform to develop . . . . .	44

4.2.	$\mu$ Trans detail over global hardware modules to develop . . . . .	45
4.3.	$\mu$ Trans 434/868 blocks diagram. . . . .	45
4.4.	MRF49XA pin diagram. . . . .	47
4.5.	$\mu$ Trans 434/868 component model for use. . . . .	48
4.6.	cNGD detail over global hardware modules to develop. . . . .	49
4.7.	Architecture model of the cNGD. . . . .	50
4.8.	PIC32 family block diagram. . . . .	51
4.9.	Prefetch cache module block diagram. . . . .	52
4.10.	Global RIIs diagram. . . . .	56
4.11.	Microcontroller to MRF49XA interface. . . . .	56
4.12.	MRF24J40MA block diagram. . . . .	57
4.13.	MRF24J40MA interface scheme. . . . .	58
4.14.	MiWi <sup>TM</sup> protocol configuration scheme. . . . .	58
4.15.	MiWi <sup>TM</sup> Protocol. . . . .	59
4.16.	ADG701 pin diagram. . . . .	59
4.17.	MCP1603 buck regulator configuration. . . . .	60
4.18.	PIC32MX675F256L primary oscillator input. . . . .	62
4.19.	PIC32MX675F256L second oscillator input. . . . .	62
4.20.	rs232SHIELD detail over global hardware modules to develop. . . . .	66
4.21.	MAX3232 basic scheme. . . . .	67
4.22.	chargerSHIELD detail over global hardware modules to develop. . . . .	67
4.23.	Collector output LM311 configuration. . . . .	68
4.24.	Configuration scheme for the LM317 provided by the manufacturer. . . . .	69
4.25.	chargerSHIELD simulation results. . . . .	70
4.26.	Specially designed footprints for the platform. . . . .	72
5.1.	$\mu$ Trans 434/868 3D model. . . . .	74
5.2.	Real size footprint and shape for $\mu$ Trans module. . . . .	75
5.3.	$\mu$ Trans antenna connectors. . . . .	76
5.4.	Detailed view of the $\mu$ Trans module. . . . .	76
5.5.	cNGD 3D model. . . . .	77
5.6.	cNGD PCB real size shape. 97x84 mm . . . . .	78
5.7.	Battery anchorage system. . . . .	78
5.8.	40-pin Header system. . . . .	79
5.9.	Detailed view of the cNGD module. . . . .	80
5.10.	rs232SHIELD 3D model. . . . .	80
5.11.	Detailed view of the rs232SHIELD module. . . . .	82
5.12.	chargerSHIELD 3D model. . . . .	82
5.13.	Detailed view of the chargerSHIELD module. . . . .	83
6.1.	cNGD current consumption at different energy modes. . . . .	86
6.2.	Spectrum energy sensing at 434 MHz. . . . .	87
6.3.	Spectrum energy sensing at 868 MHz. . . . .	88
6.4.	Spectrum energy sensing at 2.4 GHz. . . . .	88
6.5.	Batteries voltage evolution while charging with chargerSHIELD. . . . .	90

---

7.1. ICD3 programmer configuration. . . . .	94
7.2. cNGD software repository. . . . .	95
7.3. Demo application video. . . . .	98
A.1. $\mu$ Trans 434/868 schematic. . . . .	107
A.2. cNGD general schematic. . . . .	109
A.3. Microcontroller specific schematic. . . . .	111
A.4. 434 MHz Radio Interface systems schematic. . . . .	113
A.5. 868 MHz Radio Interface systems schematic. . . . .	113
A.6. 2.4 GHz Radio Interface systems schematic. . . . .	115
A.7. Power supply system schematic. . . . .	115
A.8. Expansion header system schematic. . . . .	117
A.9. Oscillator system schematic. . . . .	117
A.10. PGE system schematic. . . . .	117
A.11. rs232SHIELD schematic. . . . .	119
A.12. chargerSHIELD schematic. . . . .	119
B.1. $\mu$ Trans Top layers. . . . .	121
B.2. $\mu$ Trans Bottom layers. . . . .	121
B.3. cNGD Top layers. . . . .	125
B.4. cNGD Bottom layers. . . . .	127
B.5. rs232SHIELD Top layers. . . . .	131
B.6. rs232SHIELD Bottom layers. . . . .	131
B.7. chargerSHIELD Top layers. . . . .	133
B.8. chargerSHIELD Bottom layers. . . . .	133



# List of Tables

3.1. Comparative table of 32-bits family Microchip PICs. . . . .	38
3.2. Comparative table of sub-GHz Microchip solutions . . . . .	39
4.1. PIC32675F256L pins use description. . . . .	55
4.2. Maximum estimated power consumption of the platform. . . . .	60
4.3. Estimated values for Run and Sleep operation modes. . . . .	61
4.4. Comparison among considered batteries. . . . .	61
4.5. Headers pins use description. . . . .	65
6.1. Time costs on the communication protocol management tasks. . . . .	87
6.2. Time costs on the RIIs management tasks. . . . .	88
6.3. Effective rate, received packets comparison for 434/868 RI using P2P protocol at 115200 bps. . . . .	89
6.4. Effective rate, received packets comparison for 434/868 RI using P2P protocol at 38400 bps. . . . .	89
6.5. Effective rate, received packets comparison for 2.4 GHZ RI using P2P protocol at 250 kbps. . . . .	89
A.1. Values for balun components at different frequency bands. . . . .	107
B.1. Component description for $\mu$ Trans 434/868. . . . .	123
B.2. Component description for cNGD. . . . .	129
B.3. Components description for rs232SHIELD. . . . .	135
B.4. Components description for chargerSHIELD. . . . .	135



# Chapter 1

## Introduction

*If you hear any noise,  
hit me*

Thomas & Guy-Manuel, Daft Punk

In this chapter, a global description of the project is given. This global description covers an introduction to the context as well as the need for, and a brief description of the work. It defines goals and the project organization over time. In addition, the structure of this dissertation is described.

### 1.1. Background

A WSN (*Wireless Sensor Network*) consists of a spatially distributed number of sensors spread across a geographical area. Each sensor has wireless communication capability and some level of intelligence for signal processing and communication. WSNs provide a decentralized technological solution to address applications related to security and surveillance, industrial control, infrastructure maintenance, automatic environmental monitoring, domotics, localization, tracking, or health. They have recently emerged as a premier research topic due to their great long-term economic potential and ability to transform our lives.

Since in many applications the number of sensor nodes used will be really high or they will be placed in a remote location, accessing of the nodes might not be possible. In this case, the minimization of energy expenditure and cost is a requirement. This constraints affect the whole node operating mode, forcing the system to keep a low resources (memory, computing capability, etc.) nature.

Together with the significant growth of WSNs in the last years, it was a huge and yet increasing wireless networks and services deployment [1]. Wireless communications, specially due to wireless M2M (*Machine-to-machine*) connections, is currently one of the hottest and fastest growing segments of electronics.

The increasing demand for wireless communication, in addition to an inefficient assignment of the spectrum [2], has caused saturation on certain bands of the radio spectrum<sup>1</sup>. Some of the most heavily used bands are ISM (*Industrial, Scientific and*

---

<sup>1</sup>Radio spectrum refers to the part of the electromagnetic spectrum corresponding to radio frequencies lower than around 300 GHz.

*Medical*) bands, which are unlicensed frequencies. Most WSN solutions operate over these frequencies, like the 2.4 GHz, shared by different technologies such as Wi-Fi or Bluetooth. ISM bands are defined by region by the ITU (*International Telecommunication Union*) and local governments ultimately. Figure 1.1 illustrates the different defined regions and Figure 1.2 gives a general view of the spectrum and ISM bands over region 1.

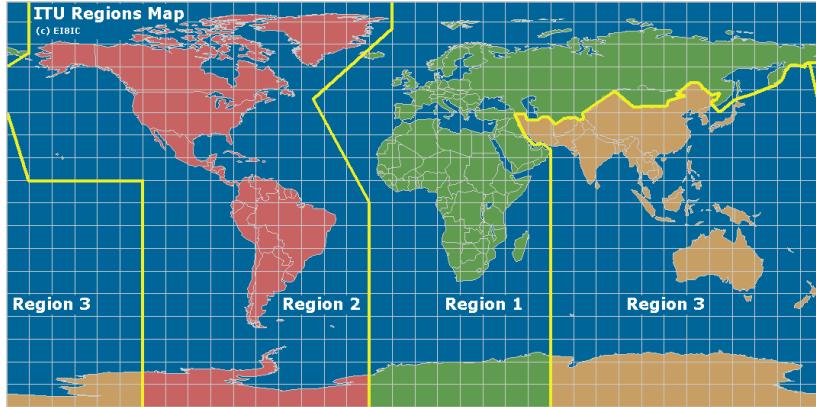


Figure 1.1: World different regions defined by the ITU, obtained from [3].

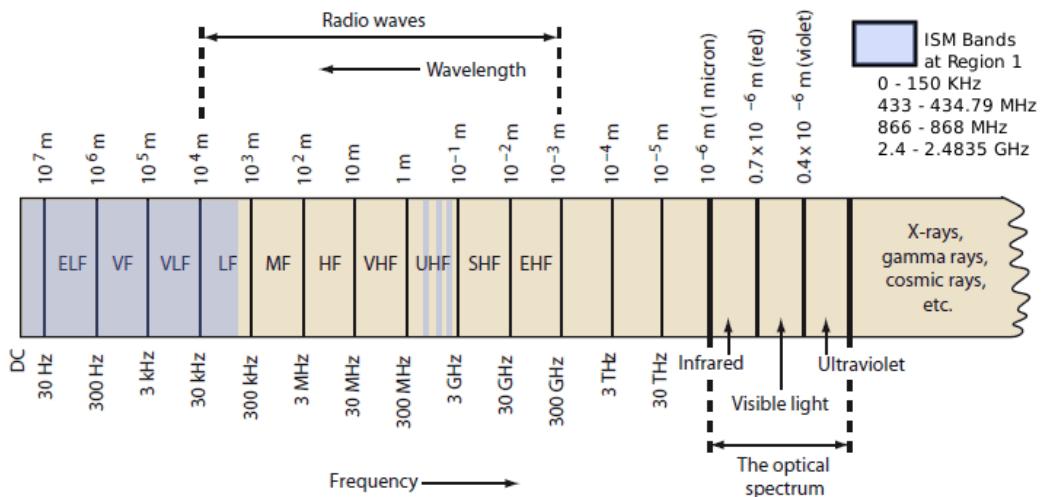


Figure 1.2: Electromagnetic frequency spectrum.

Spectrum saturation is reflected over the increasing RF (*Radio Frequency*) interferences and other undesirable effects like noise, with a consistent impact over the TX (*Transmission*) power required, QoS (*Quality of Service*), etc. Nowadays, techniques focused on expanding spectrum efficiency over communications, such as those related to modulation, power control or link adaptation, seem exhausted, and new long-term solutions are required.

To address this challenge, new techniques such as CR (*Cognitive Radio*) arise. CR was introduced by Mitola in 1999 [4] and 2000 [5]. It integrates into communication networks concepts such as spectrum sensing, self-learning, self-management, or context

analysis. Basically, CR enables opportunistic access to the spectrum through cooperation and context awareness. Spectrum sensing capabilities and cooperation among devices allow for better spectrum utilization and QoS. The concept of CN (*Cognitive Network*) describes a WN (*Wireless Network*) aware of its environment and able to adapt its communication parameters in order to achieve more reliable and efficient communications.

WSNs, among other technologies, maintain a high demand for cognitive networking. Their imposed power consumption constraints and hard operation conditions require an efficient operation mode, especially for their wireless communications, since these normally constitute the most energy-consuming task. However, there are also challenges to face [6], such as an adaptive and QoS-aware routing, or the management of control data.

Despite the potential of CWSN (*Cognitive Wireless Sensor Network*) [7], they are not deeply explored yet. Real or simulated scenarios scarcely exist, but realistic platforms are crucial to the improvement and development of this new field. The shortage and undevelopment of CWSN devices or test-beds contributes to the scarcity of results in this area.

One of the current CWSN node platforms is the FCD (*First Cognitive Device*), developed in 2011 [8] by researchers at the laboratory where this project is framed, the LSI (*Laboratorio de Sistemas Integrados*). The LSI belongs to the DIE (*Departamento de Ingeniería Electrónica*) within the ETSIT (*Escuela Técnica Superior de Ingenieros en Telecomunicación*) at UPM (*Universidad Politécnica de Madrid*). This laboratory hosts researching lines related to embedded systems, WSNs, CWSNs, or security. The implemented device became the first real generic platform to study CWSNs hosting three radio transceivers, but it was just a initial approach. The FCD was still far from being a very valuable researching test-bed platform. This platform did not fully satisfy requirements in terms of low power consumption, functionality, cost, size, and communication capabilities.

Continuing with the development of the FCD, some software implementations were carried out at LSI. A combined protocol stack for three different transceivers [9] that includes a HAL (*Hardware Abstraction Layer*) was created. This firmware offers ease of use to the researchers facing communication tasks. It also supposes memory and resources savings compared with the previous scheme, based on three protocol stacks. In order to deal with cognitive strategies and algorithmics, in addition, a software module that runs over the firmware was launched. This module, called CRmodule (*Cognitive Radio Module*) [10] implements the theoretical model described in [11] and is a crucial achievement for CWSN development.

Attending the need for devices that enable and promote CWSNs investigation, this dissertation describes the implementation and operation of the cNGD (*Cognitive Next Generation Device*), a platform for CWSN development. It includes features and capabilities not found on current devices. cNGD arises as a testbed platform for CWSNs that allows to test strategies and deploy better investigations.

## 1.2. Goals

The main goal of the project is to design and implement a node for CWSNs. The implementation must meet strong efficiency, functionality, modularity and scalability requirements. A demo application layer, also to be developed, must work integrated

with the already implemented firmware.

CWSN nodes are very limited devices in terms of memory, computational power, or energy consumption. This fact is crucial and it must be taken into account in order to make the device valuable and affordable for researching purposes.

The design must be modular, so it will not need to be entirely redone with every particular change, and adaptable, to allow flexibility in the complexity and character of applications. On the other hand, as a development platform, the software and hardware architecture must provide advantages when implementing cognitive strategies as a valuable tool (data extraction, data reliability, portability, versatility, etc.).

Hence, it is needed to reach a complete solution covering all these features.

It is important to make clear that the design must fill the gaps and deficiencies that current devices show, for example, providing a wider radio access, powering control options, or real WSN features. Furthermore, a definition of new needs, together with an important review of the FCD, are required.

Here, the main goal is fragmented into more specific subgoals:

- *Analysis of previous conclusions and results.* Study of already tested aspects. Definition of new tests to provide new data of interest. These data will help in making decisions about transceivers, microcontrollers, power supply, etc.
- *Requirements definition.* Study of needs and constraints (cost, resources, size, consumption, etc.) according to the future running applications. Definition of features and requirements. Definition of the demo application.
- *Hardware design.* Global design of the platform and new involved hardware. Development of schematics. Design optimization. Components selection and final layout deployment.
- *Hardware implementation and debugging.* Component disposition and soldering. PCB design validation.
- *Application layer development.* Full adaptation of the firmware to the hardware. Development of the demo application layer that uses the firmware and allows a full test of the platform. The demo must be a typical WSN application.
- *General purpose tests.* Tests oriented to prove the proper operation and verify the achievement of the requirements and constraints compliance.
- *Results analysis and conclusions.* Evaluation of capabilities. Study of weaknesses or possible improvements. Definition of further studies.

### 1.3. Project Organization

The organization of this project is described as follows:

- *State-of-the-art review.*

At this first stage, CWSN-related information and specific knowledge was acquired. An approach to the development tools was also outlined. Goals:

- WSN and CR state-of-the-art review. Review of the current commercial solutions.

- Adaptation to the development platform as well as other tools (workstation, repository, soldering station, hardware at the lab, etc.).
- Analysis of related projects results, either completed or under development.

■ *Hardware design and implementation.*

This phase covered from the first definitions and node specifications, to the system mounting and implementation. Goals:

- Requirements definition and general node specifications. Establishment of constraints and design criteria.
- Radio wireless interfaces and other used components evaluation (microcontroller, serial interfaces, etc.).
- Hardware full design. Selection of the components. Design adjustments and previous work verification. Layout design.
- Hardware platform implementation.

■ *Software design.*

Once the hardware started functioning, the development of software design had been undertaken. Goals:

- Full adaptation of the developed firmware to the hardware.
- Software implementation of required functions and application. Source code debugging.
- Generation of first documentation about the developed software modules.

■ *Tests and evaluation.*

Finally the proper operation of the device was evaluated and the results were analyzed. Goals:

- Fully integrated hardware and software test. Spectrum sensing, power consumption, autonomy and control, communication among nodes, connectivity to other devices, protocol stacks, etc.
- Results interpretation and conclusions review. Statement of further studies and future development lines. Found problems evaluation.

■ *Documentation generation.*

Dissertation and other required documentation (wiki, papers, manuals, etc.) writing. Dissertation will be elaborated under L<sup>A</sup>T<sub>E</sub>X[12]. Review of the software documentation.

## 1.4. Outline

In Chapter 2 takes place a whole State-of-the-art review that introduces to terms as CR, CN, WSN, or CWSN. Current hardware and software implementations are studied and main features exposed. In Chapter 3 is made a review of the necessities and gaps unattended by current devices. Requirements and first conclusions to face the design are obtained. Chapter 4 covers the design process. Schematics, used components

and design decisions are covered in detail. Hardware implementation, PCB (*Printed Circuit Board*) making and mounting are described at Chapter 5. Carried out tests and results are exposed at Chapter 6, whereas the Chapter 7 covers all the software developments. Chapter 8 resume some conclusions and future lines, and Chapter 9 provides the project estimated costs.

## Chapter 2

# Cognitive Wireless Sensor Networks: State-of-the-art

*Television rules the nation*

Thomas & Guy-Manuel, Daft Punk

This chapter shows an approach to the fundamentals of Wireless Sensor Networks and the current development state of new paradigms such as Cognitive Radio, Cognitive Networks and Cognitive Wireless Sensor Networks. It describes current related devices, applications and implementations, and introduces terms used throughout this dissertation.

### 2.1. Cognitive Radio

Nowadays, it is commonly believed that there is a crisis of spectrum availability for wireless communications [13]. However, according to regulatory bodies as the FCC (*Federal Communications Commission*) [14] or Ofcom (*Office of Communications*), most of the radio frequency spectrum is under-utilized while some spectrum bands are heavily used. Military, amateur radio or satellital frequencies, for instance, are insufficiently utilized compared to cellular networks or the overcrowded ISM bands [2].

Most of the spectrum is allocated to specific applications and the static assignment of the spectrum results in an inefficient use of it. Figure 2.1 shows how the actual utilization in the 3-4 GHz frequency band is 0.5 % and drops to 0.3 % in the 4-5 GHz band. This seems totally in contradiction to the concern of spectrum shortage.

Spectrum utilization depends strongly on time and place, however, fixed spectrum allocation prevents specific assigned frequencies from being used, even when this use would not cause noticeable interference to the assigned service. These facts lead to the current inefficient situation where the utilization of the total spectrum can be considered around 10 % and more than 95 % of the use is below 3GHz.

The concept of CR was first published by Joseph Mitola III and Gerald Q. Maguire, Jr. in 1999 [4] and later on within Mitola's PhD dissertation in 2000 [5]. It describes a novel paradigm for wireless communication in which a wireless device changes its transmission or reception parameters to communicate efficiently. This alteration of parameters is based on the active monitoring of several factors in the external and internal radio environment, such as radio frequency spectrum, application behavior,

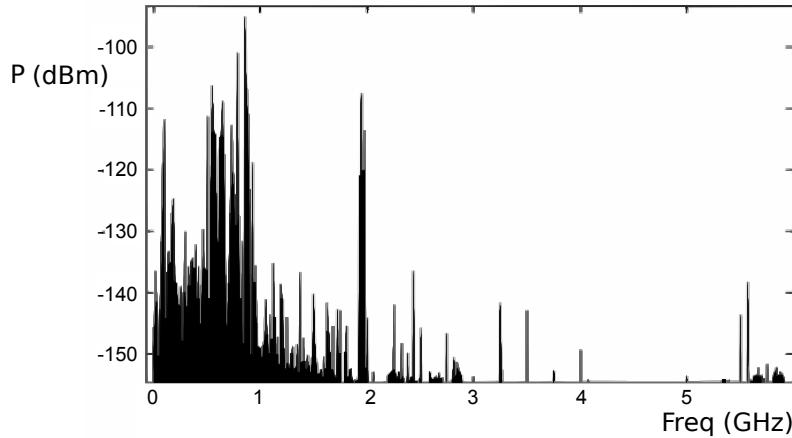


Figure 2.1: A snapshot of the spectrum utilization up to 6 GHz in an urban area: taken at mid-day with 20 kHz resolution over a time span of 50 microseconds with a 30 degree directional antenna at Berkeley Wireless Research Center [15].

and network state. The idea was thought of as an ideal goal towards which an SDR (*Software-Defined Radio*) platform should evolve.

CR is defined as an intelligent wireless communication system that is aware of its environment and uses the methodology of understanding-by-building [16] to learn from the environment and adapt to statistical variations in the input stimuli, with two primary objectives:

- Highly reliable communications.
- Efficient utilization of the radio spectrum.

Although the concept of CR was defined originally as an extension to SDR [4], which is able to reason about external factors, afterwards, the term has mostly been used in a narrower sense. FCC suggests in [17] that any radio having adaptive spectrum awareness should be referred to as CR:

*“A CR is a radio that can change its transmitter parameters based on interaction with the environment in which it operates... The majority of cognitive radios will probably be SDRs, but neither having software nor being field programmable are requirements of a cognitive radio.”*

Considering some subtle differences between systems we can differentiate two main different types of CR:

- *Spectrum-Sensing Cognitive Radio*, in which only the radio-frequency spectrum is considered [16].
- *Full Cognitive Radio or Mitola radio*, in which every possible parameter observable by a wireless node is considered [4].

Although cognitive radio was initially thought of as full cognitive radio, most research work focuses on spectrum-sensing cognitive radio, particularly in the TV

bands. The great problem in spectrum-sensing cognitive radio is designing high-quality spectrum-sensing devices and algorithms for exchanging the so-called knowledge domain [16]. The practical implementation of spectrum-management functions is a complex and multifaceted issue, since it must address a variety of technical and legal requirements.

Picture 2.2 illustrates the key functions of the PHY (*Physical*), MAC (*Medium Access Control*), and network layers in a CR. This model, described in [18], supposes a functional architecture that is by no means the only architecture that can be designed for CRs.

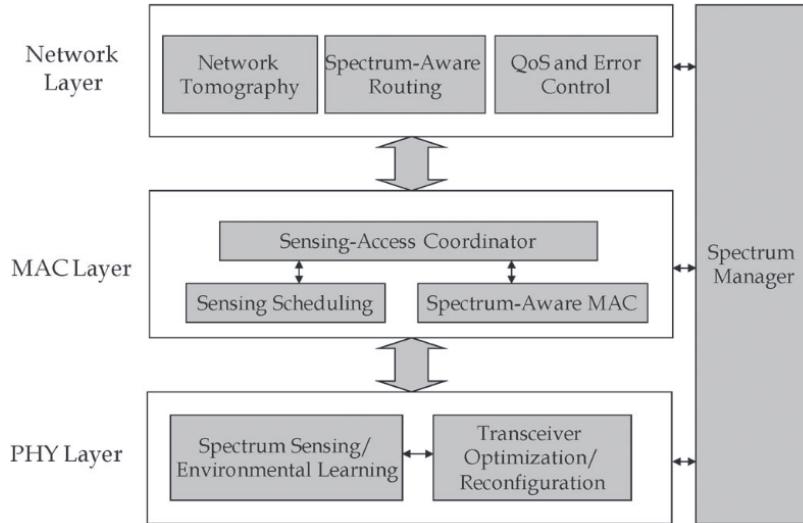


Figure 2.2: Main functions of the PHY, MAC, and network layers in a CR, obtained from [18].

Depending on the parts of the spectrum used, CR can be:

- *Licensed-Band Cognitive Radio*, capable of using bands assigned to licensed users. These licensed users are known as primary users.
- *Unlicensed-Band Cognitive Radio*, which can only utilize unlicensed parts of the RF spectrum. One such system is described in the IEEE (*Institute of Electrical and Electronics Engineers*) 802.15 Task Group 2 specifications [19].

The main functions of CR devices are: [20][21]

- *Spectrum sensing*: An important requirement is detecting unused spectrum and sharing it, without causing interferences to other users; different categories of spectrum-sensing techniques might be distinguished:
  - *Transmitter detection*: CR must have the capability to determine if a signal from a primary user is locally present in a certain spectrum. Enclosed here we can find that approaches such as *matched filter detection*, *energy detection* or *cyclostationary-feature detection* are common.
  - *Cooperative detection*: Refers to spectrum-sensing methods where information from multiple CR users is integrated [22].

- *Interference-based detection.* It tries to take advantage of the difference between the detected noise level and the maximum noise level the primary users can afford. In this way, if there is enough difference, the nodes might establish RF activity noticeable by primary users as simple noise. This technique is not so common because of disagreements with primary users.
- *Power Control:* Power control is used for both, opportunistic spectrum access, and proper CR spectrum sharing. On the one hand, opportunistic spectrum access techniques aim to find the cut-off level in SNR (*Signal to Noise Ratio*) supporting the channel allocation. On the other hand, CR systems find imposed interference power constraints for the primary user's protection. In [23] a joint power control and spectrum sensing is proposed for capacity maximization.
- *Spectrum management:* Capturing the best available spectrum to meet user communication requirements, while not creating undue interference to other users. CR should decide on the best spectrum band (over the available range) to meet QoS requirements; therefore, spectrum-management functions are required for CR. Spectrum-management functions are classified as *Spectrum sensing* and *Spectrum decision*.

Realizing that CR technology has the potential to exploit inefficiently utilized licensed bands without causing interference to incumbent users, the FCC released a Notice of Proposed Rule Making which would allow unlicensed radios to operate in the TV-broadcast bands. The IEEE 802.22 working group [24], formed in November 2004, is tasked with defining the air-interface standard for wireless regional area networks (based on CR sensing) for the operation of unlicensed devices in the spectrum allocated to TV service [25].

Dynamic spectrum allocation has become a key research activity in the wireless communications field and in particular a key technology for “The Network of the Future”, objective proposed in ICT FP7 (*7th Framework Programme for Research and Technological Development*)<sup>1</sup>.

## 2.2. Cognitive Networks and Cognitive Radio Networks

Over recent years, “cognitive” or “smart” have become *trending topics* being applied to many fields, including communication technologies. Looking at some recent literature, it is easy to find mentions of smart antennas [26], smart radios [27], smart packets [28], CR [4][16], cognitive packets [29] and CN [30][31]. Nevertheless, there does not seem to be a commonly accepted definition of what these terms mean when applied to networking technologies.

The concept of CN has been in the collective psyche of the networking and wireless research field for long. The first approach was made by Mitola [4], who briefly described how the CR could interact within the system-level scope of a CN. Saracco [32] talked about CNs in his investigation into the future information technology. Mähönen et al. [30] discussed CNs with respect to future mobile IP (*Internet Protocol*) networks. None of these previous references, however, expressed clearly what a CN is, how it should work and which problems it should solve.

---

<sup>1</sup>Funding programmes created by the European Union in order to support and encourage research in the European Research Area.

The role that CR had in inspiring the formulation of CN concept caused, in some cases, CNs to be described as networks of CRs [4][33]. Recent research could be divided into two categories: CRN (*Cognitive Radio Network*) and CNs.

For CRN, Mitola mentions how CRs could interact within the system-level scope of a CN [5]. Neel [34] and Haykin [16] continue this line of thinking, examining multi-user networks of CRs as a game. The scope of CRNs still remains primarily on MAC and PHY layers, but now operating with some end-to-end objective. In a CRN, the individual radios take most of the cognitive decisions, although they may act in cooperation. Some suggested applications for CRNs include cooperative spectrum sensing [35][36] and emergency radio networks [37]. Raychaudhuri presents in [38] a general architecture for CRNs.

Regarding CNs, Clark proposes [39], in what was perhaps the first mention of CN rather than CRN, a network that can

“assemble itself given high level instructions, reassemble itself as requirements change, automatically discover when something goes wrong, and automatically fix a detected problem or explain why it cannot do so.”

This would be achieved with the use of a KP (*Knowledge Plane*) that transcends layers and domains to make global cognitive decisions. The KP would add intelligence and weight to the edges of the network, and context sensitivity to its core. Saracco stated [32] that the change from network intelligence controlling resources to having context sensitivity would help *flatten* the network by moving network intelligence into the core and control further out to the edges of the network. CNs differ from CRNs in that their action space extends beyond the MAC and PHY layers and the network may consist of more than just wireless devices.

The first full definition of CN was postulated by Thomas [40] in his PhD thesis. He proposed the idea of a CN:

“*a network composed of elements that, through learning and reasoning, dynamically adapt to varying network conditions in order to optimize end-to-end performance. In a CN, decisions are made to meet the requirements of the network as a whole, rather than the individual network components.*”

The adaptations that are performed over usual networks are commonly reactive, taking place after a problem has occurred. Thomas advanced a paradigm that had the promise to remove these limitations by allowing networks to observe, act, and learn in order to optimize their performance. The description of CNs in [40] talks about intelligently selected and adapted radio spectrum, transmission power, antenna parameters and routing tables. By formalizing the design, architecture and tradeoffs of cognition at the network level, Thomas’s work had a broad impact on advancing the paradigm of intelligent communication devices.

### 2.2.1. A simple example

This example was published in [40] and it is inspired and influenced by Daniel Friend’s example in [41]. It illustrate the need for end-to-end rather than link adaptations.

Consider an ad-hoc data session between a source node  $S_1$  and a destination node  $D_1$  as shown in Figure 2.3. The source node does not have enough power to reach  $D_1$

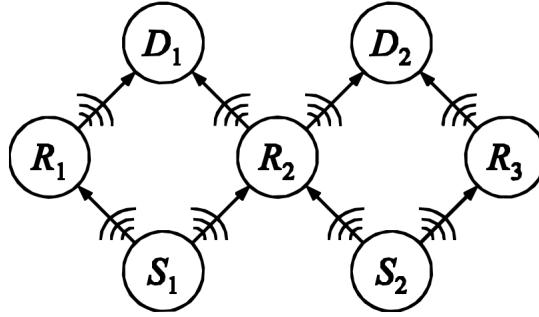


Figure 2.3: Simple relay network for a wireless network, vertex represent wireless connectivity, obtained from [40].

directly, so it must route traffic through intermediate nodes  $R_1$  and/or  $R_2$ . Assume that the end-to-end goal is to have the highest probability of successful transmission. The routing layer will determine routes based on minimum hop count which, in this case, includes either  $R_1$  or  $R_2$ . Node  $S_1$  will make a link-layer adaptation, selecting between  $R_1$  and  $R_2$  based on their SINR (*Signal to Interference and Noise Ratio*). From the standpoint of the link layer in node  $S_1$ , this ratio correlates with the probability that the transmitted packets will arrive correctly at the relay node. However, without additional information, this selection does not guarantee anything about the end-to-end packet delivery probability from  $S_1$  to  $D_1$ . In contrast to a link adaptation, the CN might use some combination of observations from all nodes to compute the total path outage probabilities from  $S_1$  to  $D_1$  through  $R_1$  and  $R_2$ . This shows the benefit of an end-to-end scope, but there is another advantage for the CN, its cognitive capability. To illustrate this, we modify the original scenario to include both  $S_1$  and  $S_2$  as source nodes, both routing traffic through  $R_2$ . Suppose that the learning mechanism measures outages by determining the fraction of packets successfully delivered from the source to its destination.

If  $R_2$  becomes congested because of a large volume of traffic coming from  $S_2$ , this becomes apparent to the cognitive process by the lack of successful packet delivery statistics provided to  $S_1$  and  $S_2$ . The learning mechanism recognizes that the system has changed and that routing through  $R_2$  is not optimal. The cognitive process then directs the traffic toward another route. The CN does not explicitly know that there is congestion at node  $R_2$  because this information is not included in the SINR observations. Nevertheless, it is able to infer from the reduced throughput that there may be a problem. It is then able to respond to the congestion, perhaps by routing traffic through  $R_1$  and/or  $R_3$ . This example shows the power of CNs in optimizing end-to-end performance as well as reacting to unforeseen circumstances.

### 2.3. Wireless Sensor Networks

A WSN consists of spatially distributed autonomous “nodes” not relying on a pre-existing communication infrastructure, to monitor physical or environmental conditions. The number of nodes vary from a few to several hundreds or even thousands, where each node is connected to one (or sometimes several) sensors. Nodes are generally simple and low-resource embedded systems with high cost and size constraints.

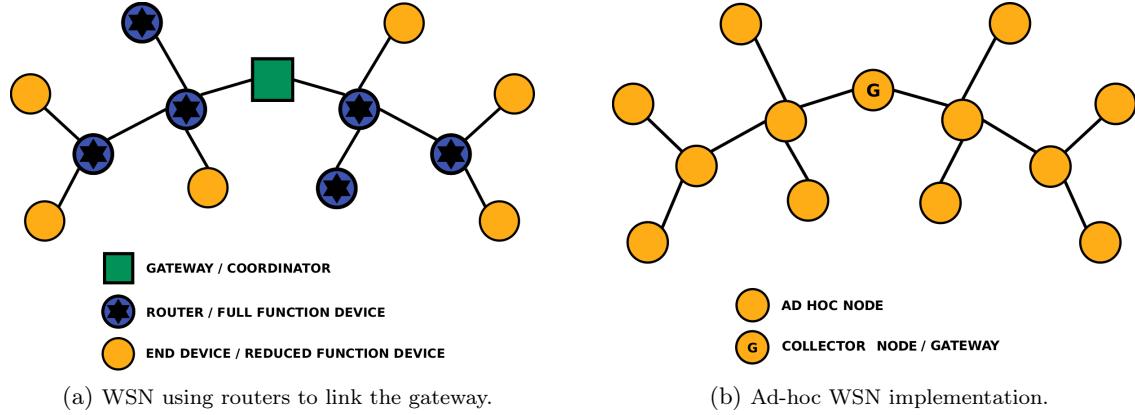


Figure 2.4: Possible WSN implementations, obtained from [42].

These constraints result in corresponding constraints on resources such as energy, memory, computational speed and communications bandwidth. The usual architecture of a node is divided in:

- *Sensing subsystem*. Responsible for sensing physical parameters of the environment. Typical monitored parameters are temperature, sound, light, humidity, vibrations, pressure, movement, presence, and body registers.
- *Computational subsystem*. It process the information obtained by the sensors, controls all general operations of the node, and runs the desired application.
- *Communication subsystem*. This carries through all the messages transmission and reception with neighbor nodes. The main goal is to make the information arrive to some destination, usually a gateway or storing node.
- *Power source*. It supplies the energy needed by the device to perform the programmed task.

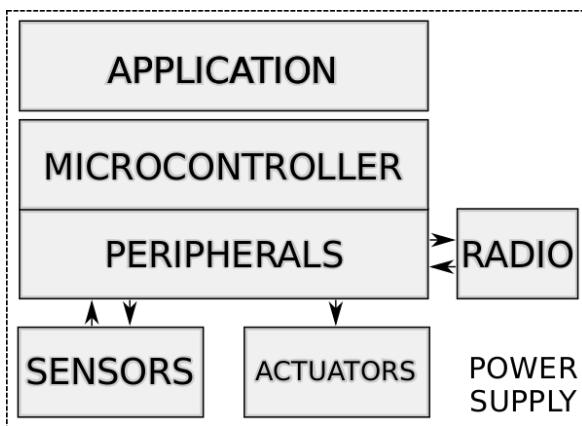


Figure 2.5: WSN node general model

The development of wireless sensor networks was motivated by military applications such as battlefield surveillance; today such networks are used in a wide range of

applications and they are considered to be the main technology to develop intelligent ambiances:

- *Industrial Monitoring.* WSN are applied mainly for machine health monitoring, water quality or management, and industrial sense and control applications avoiding wire deployment.
- *Structural Health Monitoring.* Distributed sensors over infrastructure such as bridges, tunnels or buildings help to collect data to prevent damages or problems deviated from load excess, weather, vibrations or stress.
- *e-Health.* It presents new uses for WSN to sense body parameters and observe behavioral patterns. These networks are used to detect or prevent occupational and home accidents, to improve diagnoses, monitoring sick people and other medical uses.
- *Environmental monitoring.* Applications regarding this field are diverse. Air quality monitoring, air pollution monitoring, forest fire detection, landslide detection, water quality monitoring or natural disaster prevention are some of the common uses.
- *Smart home monitoring.* Monitoring the activities performed in a smart home is achieved using wireless sensors embedded within everyday objects. State changes are captured by the WSN enabling activity-support services.
- *Passive localization and tracking.* Applications oriented to detect where something took place or track presences over an area.
- *Agriculture.* Commonly used on greenhouses where irrigation management and ambient control is essential for a proper accurate agriculture.
- *Security and surveillance.* Deployed to detect unauthorized intrusions.

Some of the main properties affecting WSN are:

- *Dynamic topology.* In a WSN it is common to suffer drops or rises in the number of nodes, or changes over the environment that affects the topology. Nodes must be able to adapt themselves to new topologies to enable operative communications. On the same way, topologies must be scalable since a network might have tens of nodes or hundred of them, in some cases even mobile ones.
- *Autonomous operation.* A WSN does not need any infrastructure to operate. Its nodes act as information transmitters, receivers or routers. However, it usually exists a gateway which gathers all the information over the network and passes it to another device.
- *Multihop or broadcast communications.* It is common the use of some protocol to enable multi hop messaging. Nevertheless, broadcasting is also very expanded.
- *Power consumption.* One of the most important factors. Using a very constrained amount of energy, devices must achieve a tradeoff between autonomy and throughput. Normally, nodes are supplied with batteries which allow their autonomy. Hence, a WSN node must meet a low-consumption microcontroller, as well as an equally featured RI (*Radio Interface*) and software.

- *Hardware constraints.* In order to achieve a low power consumption and cost, it is essential for the hardware to be as straightforward as possible, resulting in a very constrained functionality.
- *Size.* Nodes tend to be small in size since it is used to be needed a large amount of them. Most of the time, they are intended to be encrusted into the environment and to be transparent to the users.
- *Production costs.* Since WSNs nature implies having a high number of nodes to be trustable, production of large amounts of them must provide a cheap unitary price.

Most important communication technologies and protocols for WSN are based on the IEEE 802.15.4 [43] standard for WPAN (*Wireless Personal Area Network*). The standard goal is a low-power communication among nearby devices without underlying infrastructure. The standard only defines MAC and PHY layers of the OSI (*Open Systems Interconnection*) model. Some expanded protocols based on the standard are ZigBee<sup>TM</sup> [44], WirelessHART [45], ISA100.11 [46], or MiWi<sup>TM</sup> [47] specifications. Another popular communication standard is IEEE 802.11 [48], in which is based Wi-Fi [49]. It is a IP standard based on the final user and does not meet low-consumption constraints, however, convergence towards full IP has brought new standards such as 6LoWPAN (*IPv6 Low Power WPAN*) [50], that enables IP packeting over 802.15.4 based-on networks.



Figure 2.6: Common WSN protocols.

## 2.4. Cognitive Wireless Sensor Networks

Basically, CR and CN techniques applied into WSNs leads to CWSNs. This idea supposes an increment of complexity over the executed algorithms and it overloads the control data flow over the network. Cognitive agents capable of making proactive decisions may help achieve end-to-end goals, while CR at the physical layer of such agents may enable the opportunistic use of the heterogeneous wireless environment.

CWSNs arise as a natural evolution of traditional WSNs since IEEE standards used in WSNs already postulate access to several spectrum bands. Additionally, QoS and low-consumption requirements that WSNs state, fits the goals of CR.

Currently, most of WSN use the IEEE 802.15.4 standard for their communications, thereby their RF activity takes place over ISM bands. These bands, as introduced in Section 1.1, show an overcrowded scene where CR techniques might significantly help the network operation. Applying cognitive capabilities seeks intelligent adaptations based on learning, reasoning and information sharing among multiple nodes within the network.

Cognitive communication in a sensor network could not only help meet end-to-end goals of the entire network, but also increase reliability of the network, reduce maintenance costs and increase the network lifetime. Research in [51][52][53][54] suggests the growing interest in applying cognitive techniques to WSNs. The idea of a holistic approach to introducing cognition in heterogeneous sensor networks that combines the advantage of opportunistic spectrum access at the physical layer, with cognitive communication among sensor nodes seamlessly across the network promises to be advantageous over existing design techniques. Figure 2.7 shows a performance comparison between a standard WSN communicating over 2.4 GHz and a CR-based one over a simulated scenario.

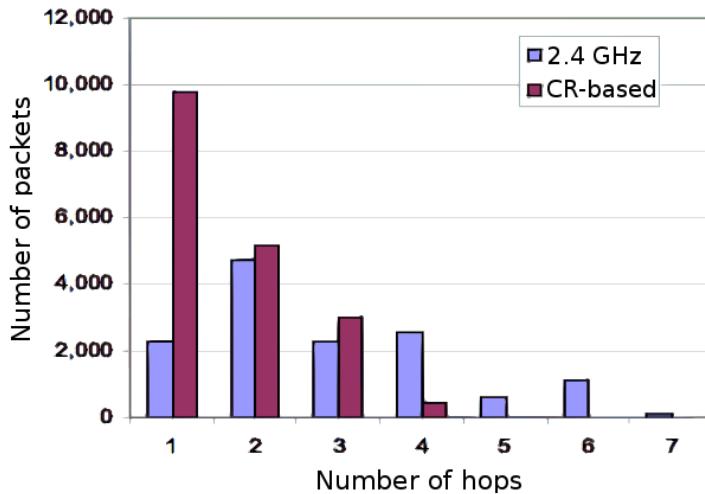


Figure 2.7: Comparison of the number of hops per packet in a CWSN simulated scenario, described in [54].

However, research efforts have been discrete and cognitive techniques have focused on improving specific aspects of the network or benefiting specific applications.

Vijai contributes [7] to CWSN giving the vision and advantage of a holistic approach to cognition in sensor networks. It also provides a framework based on knowledge and cognition.

In [11], opportunities and trends arising from networks and nodes cooperation are mentioned. It is seen as a chance to improve general features and dynamic adaptation capabilities. It proposes an implementation model based on agents carrying out basic functions and keeping a VCC (*Virtual Control Channel*). This VCC takes part on the KP that agents use to exchange management messages.

On the other hand, applying cognitive techniques to increase knowledge in the system has several challenges:

- Establishing the feasibility of integrating CR into the DSA (*Dinamic Spectrum Access*) [20] scheme at the physical layer, along with cognition in upper layers to achieve end-to-end performance goals is an open research problem.
- For such networks to be justifiable, the performance improvement must outweigh the cost in terms of overhead, architecture, and operation. An analysis on the amount of energy expended at information sensing and communicating it to neighboring nodes is essential to establish the suitability of this approach.
- Since the available information to the network may be partial or incorrect, it may lead to security issues and hence, methods and techniques to deal with such issues must be also identified.
- The proposed cognitive nodes could be distributed in a fix or mobile way in the network, gathering information from remote locations of the sensor nodes. Hence, deciding on the optimal deployment architecture of the cognitive capability is also a challenging problem.
- Protocols that define how the knowledge plane can be implemented, how to use it to make decisions at the physical layer, the cognitive specification language, and the tools used in cognitive decision making must all be standardized to make such networks interoperate.

All in all, this early-staged technology claims for investigations and research-enabling implementations such as standards, simulators and test-beds that allow further studies and conclusive results.

#### 2.4.1. Current Implementations

Because of the novel stage of this research field, there are not many specific devices to build applications and services over CWSNs. Besides, current implementations find themselves very poorly featured yet, not responding researchers' requirements. It is natural that most works are based differently on WSN and SDR.

On the one hand, there are many different types of devices for WSN platforms that share similar characteristics: low power, memory and processing constraints, and operation over ISM bands. Bean, BTnode, MANTIS Nymph, IMote, MicaZ, Sense-Node, XYZ, Sentilla Mini, TelosB [55], ANT [56], and EyesIFX [57] are some of the most important WSN devices. But none of them have different RIs and their radio reconfiguration capabilities are very limited.

On the other hand, many SDR platforms have been developed to support individual research projects. Berkeley Cognitive Radio Platform [58] (based around the BEE<sup>2</sup><sup>2</sup>), OpenAirInterface [59] (proposed by the mobile communications department at EURECOM), NICT<sup>3</sup> SDR platform [60], the Kansas University Agile Radio (KUAR) [61], or the USRP (*Universal Software Radio Peripheral*) [62] are the most important ones.

In order to evaluate CWSN models and architectures, great efforts developing simulators or adapting traditional simulators to new schemes took place. Nevertheless,

---

<sup>2</sup>Berkeley Emulation Engine system is designed to be a modular, scalable FPGA-based computing platform

<sup>3</sup>Japanese National Institute of Information and Communications Technology

test-beds allow to test real systems, obtaining data about consumption, radio transmission ranges, error rates and providing trustable feedbacks to improve simulators performance. Here it is made a review over existing software and hardware platforms.

#### 2.4.1.1. Software platforms - simulators

Some of the most representative simulators are:

- *NS2/NS3* [63]. NS (*Network Simulator*) is a discrete-event network simulator primarily used in teaching and research. It supports a large variety of multicast and unicast protocols for both wireless and wired networks. Intensively used in wireless mobile ad-hoc networks. NS simulators are developed mainly under C++ and Python languages and they are publicly available under the GNU GPLv2 license for research, development, and use.



(a) NS2 simulator logo.



(b) NS3 simulator logo.

Figure 2.8: NS2 and NS3 simulators logo.

Currently, NS2 source code consists of many forks. Last version, from 2009, is partially maintained. It runs on GNU/Linux, FreeBSD, Solaris, Mac OS X and Windows 95/98/NT/2000/XP. NS3 would be written from scratch, not being compatible with NS2 generally. Development of NS3 began in July 2006. The first release, NS 3.1, was made in June 2008. Afterwards, the project continued making quarterly software releases, and more recently has moved to three releases per year. NS3 made its fifteenth release (ns-3.15) during the third quarter of 2012 and it is actively developed.

Libraries and packages for cognitive simulation [64] exist for these simulators, however, models, parameters, and results are still quite poor and inaccurate.

- *OMNeT++* [65]. It is not a simulator itself but rather an extensible, modular, multi-platform, component-based C++ simulation library and framework. Instead of containing a real time cognitive radio environment for physical and link layer experiments and explicit support for computer networks or other areas, it provides the infrastructure for writing such simulations. OMNeT++ provides a component architecture for models. Components (modules) are programmed in C++, then assembled into larger components and models using a high-level language (NED).



Figure 2.9: OMNeT++ simulator logo.

These models, most of them open source, are developed completely independently of OMNeT++, and follow their own release cycles. They cater for domain-specific functionalities, such as support for sensor networks, wireless ad-hoc networks, Internet protocols, performance modeling, photonic networks. There are extensions for real-time simulation, network emulation, alternative programming languages (Java, C#), database integration, SystemC integration, and several other functions.

- *MiXiM* [66]. It is an OMNeT++ modeling framework created for mobile and fixed wireless networks (WSNs, body area networks, ad-hoc networks, vehicular networks, etc.). It offers detailed models of radio wave propagation, interference estimation, radio transceiver power consumption, and wireless MAC protocols. It is a merger of several OMNeT++ frameworks written to support mobile and wireless simulations. The predecessors of MiXiM are ChSim [67], Mobility Framework [68], Mac Simulator and Positif Framework [69].
- *Castalia* [70]. It is a WSN simulator based on OMNeT++ for early-phase algorithm/protocol testing built at the Networks and Pervasive Computing program of National ICT Australia since 2006. In 2007 it is made public as an open source project under the APL (*Academic Public License*) license. The current release version is 3.2. It supports realistic channel and radio models, a key element for accurate early-phase WSN simulation. It provides support for defining versatile physical processes for specific applications, since it is highly parametric, and can simulate a wide range of platforms. It also supports enhanced modeling of the sensing devices and other often-neglected attributes of a WSN such as node clock drift<sup>4</sup>.



Figure 2.10: Castalia simulator logo.

CWSN support in castalia was first proposed and developed by researchers from LSI at UPM in 2012 [71].

- *NetSim* [72]. NetSim, firstly released in June 2002, is a stochastic discrete event simulator developed by Tetcos in association with Indian Institute of Science. This popular network simulation tool is used for network lab experimentation and research. Various technologies such as WSN, Wireless LAN, WiMax, TCP, or IP are supported. NetSim comes with an in-built development environment, which serves as the interface between user's code and NetSim's protocol libraries and simulation kernel. Protocol libraries are available as open C code for user modification.

Some libraries available for NetSim enables CR capabilities for simulations. However, these cognitive libraries are focused on applications for 802.22 WRAN ba-

---

<sup>4</sup>Clock drift refers to several related phenomena where a clock does not run at the exact right speed compared to another clock.



Figure 2.11: NetSim simulator logo.

sed cognitive radio networks, which make this simulator undesirable for CWSN scenarios.

- *SENDORA (Sensor Network for Dynamic and Cognitive Radio Access)* [73]. SENDORA project developed in 2010 supposed a new approach of CR called Sensor Network aided Cognitive Radio. This project was led by Thales, Eurecom, NTNU, Telenor, KTH, TKK, Universities of Rome, Valencia and Linköping. It was divided into 8 work packages that covered from management activities to dissemination, passing through definition, integration, implementation and demonstrations activities. Developed software was based on the NS simulator and hardware implementations operated over VHDL language.



Figure 2.12: SENDORA simulator logo.

The SENDORA project brought a high amount of papers and literature, nevertheless, the software developed has come shifted to the background because of other simulators. Regarding hardware implementations, carried out over FP-GA (*Field Programmable Gate Array*), revealed useful data but not realistic for WSNs. Hence, simulations deployed do not use real device data for power models.

Several other simulators have been developed for WSN. TOSSIM based on the TinyOS<sup>5</sup> operative system, COOJA, OPNET, GloMoSim, JSim, NetSIm, or QualNet, all of them are simulators without cognitive features despite some approaches and efforts to create frameworks enabling them.

#### 2.4.1.2. Hardware platforms

Nowadays there are not real devices for CWSN applications yet. Current implementations respond to development platforms or test-beds, and still, variety of platforms is very scarce and poorly featured. Most of them suppose first approaches, so foundations to build over are still immature and quickly changing. Hence, efforts focused on hardware development, usually more costly than software, remain quiet.

The closest existing device to a CWSN node is the FCD [8][74], illustrated at Figure 2.13, developed at LSI in 2011. The FCD is based on a PIC32 Microchip MCU (*Microcontroller*) and includes three RIs enabling access to the 2.4 GHz and 868 MHz bands. MiWi and Wi-Fi protocols operate over 2.4 GHz and a proprietary protocol

---

<sup>5</sup>TinyOS is a free and open source software component-based operating system and platform targeting WSNs).

provided by AWD (*Advanced Wireless Dynamics*)<sup>6</sup> operates over 868 MHz. The device gave the chance to develop and test algorithms, strategies and applications for CWSNs. Moreover, it allowed to analyze suitability for RIs, computing capability and other components. It gave the strengths and weaknesses to establish the fundamentals on future designs. Nevertheless, the device was just a first approach and it was still far to be a stable valuable test-bed platform. The node did not fully satisfy requirements in terms of low power consumption, cost and size, and communication capabilities [9].

Consecutively, new hardware and software modules were incorporated to expand its features. These modules included a expansion board [10] to try new transceivers together with an optimized CWSN oriented firmware and a cognitive software module. This new modules claim for a new design integrating together all of them at one single module, fixing detected weaknesses and adding new or improved features.

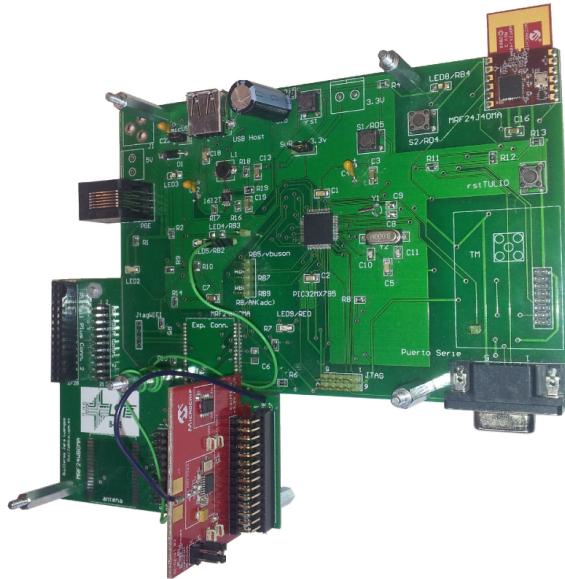


Figure 2.13: Picture of the expanded First Cognitive Device, obtained from [9].

On the other hand, some other existing devices that share common features with standard WSN nodes have supposed an approach to CWSNs. The most important devices whose features frame them into the CWSNs are here described:

- **VESNA** [75]. It is a modular and fully flexible platform, implemented by SensorLab at Jozef Stefan Institute [76], for WSN development. Based on a high-performance microcontroller with ARM Cortex-M3 core and RI spanning over multiple ISM frequency bands, it is designed to meet the requirements of diverse applications. In terms of modularity, the platform consists of the VESNA core module and a set of special feature modules (sensor node radio - SNR, sensor node expansion - SNE, sensor node power - SNP) that are used as/if needed. Various peripherals including UART (*Universal Asynchronous Receiver-Transmitter*), I<sup>2</sup>C (*Inter-Integrated Circuit*), SPI (*Serial Peripheral Interface*), USB (*Universal Serial Bus*), ADC (*Analog-to-Digital Converter*) and DAC (*Digital-to-Analog Con-*

---

<sup>6</sup>A spin-off originated at LSI, oriented on solutions for energy efficiency, industrial control and intelligent environments.

verter) allow hosting of different sets of sensors and/or actuators. The platform readily supports:

- Communication standards IEEE 802.15.4, IEEE 802.15.1 and IEEE 802.11.
- ZigBee, 6LoWPAN, Bluetooth and Wireless M-Bus protocol stacks and technologies.
- Operating system Contiki<sup>7</sup>.
- Connection to the Internet via Wi-Fi, Ethernet or GSM/GPRS;
- Arduino integrated development environment.
- A variety of energy supply options including battery, solar panel and external power supply.

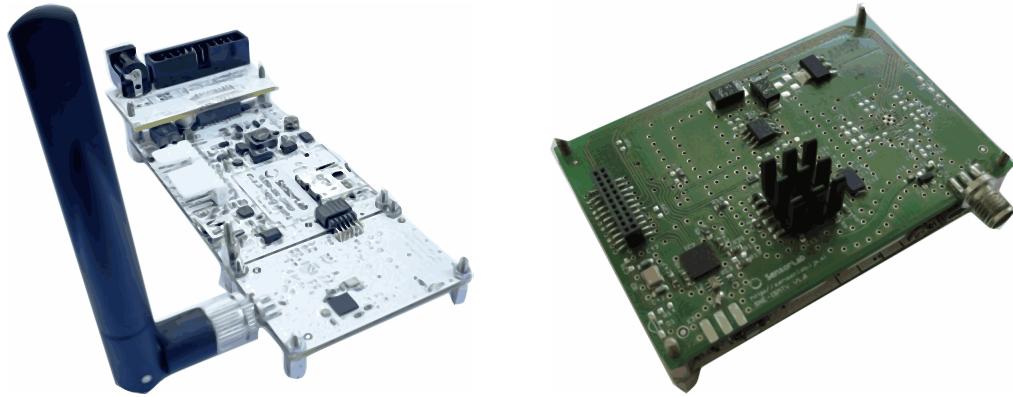


Figure 2.14: Pictures of VESNA and SNE-ISMTV modules.

To enable CR capabilities on VESNA, a radio front-end was developed, the SNE-ISMTV [77]. Different versions were capable of operating at three frequency bands of interest (TV broadcast part of the UHF band, 868 MHz and 2.4 GHz ISM band) and flexible enough to enable various user scenarios.

SNE-ISMTV-UHF contains a VHF and UHF TV band receiver based on the NXP TDA18219HN silicon tuner and was designed for spectrum sensing experimentation in TV white spaces. SNE-ISMTV-UHF can receive signals from 470 to 870 MHz with tunable channel filters. Using an analogue detector it can be used for energy detection experiments. The detector is also coupled with an ADC optionally providing digital signal processing.

SNE-ISMTV-868 and SNE-ISMTV-2400 are based on the TI CC1101 and TI CC2500 integrated circuits respectively and are identical in design and operation except for the supported frequency band. These transceivers contain software-reconfigurable radio front-ends. They include an integrated energy detector and several modems that enable different signal processing modes.

This platform does not offer frequency agility over several frequency bands since it just hosts possibilities for a single RF front-end and this front-end supports fixed bands. It does not suppose a complete CWSN solution.

---

<sup>7</sup>Contiki is an open source operating system for networked, memory-constrained systems with a particular focus on low-power wireless Internet of Things devices.

- *WaspMote*. It is an ultra low-power consumption sensor node developed and marketed by Libelium<sup>8</sup> based on an ATmega1281 MCU. It provides a RI socket and several standardized input/output options. Libelium offers up to 60 different sensor expansion boards and 6 different radio options. An attachable board enables connection for two RIs. It is thought to be a versatile node valid for a wide range of WSN applications.

WaspMote, however, does not give chances for more than two RIs. Frequency agility is, hence, bounded to two spectrum bands. Moreover, it does not give software support for easy radio management and application development. It still supposes a poor CWSN solution.

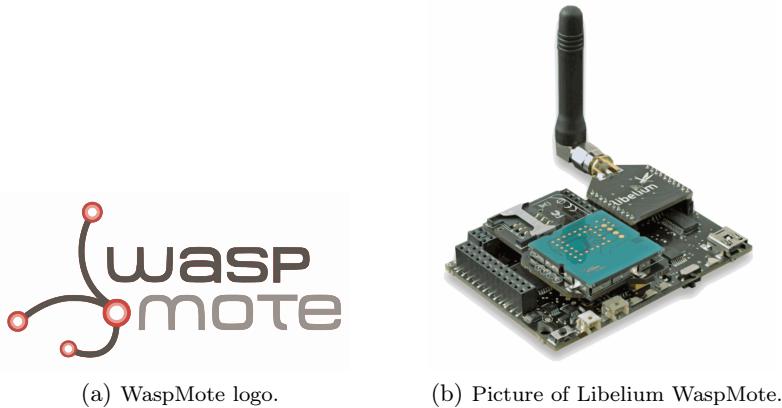


Figure 2.15: Libelium WaspMote.

- *Meshlum*. Meshlum, also from Libelium, works as the gateway for WaspMote SN (*Sensor Network*). It reads the sensor frames coming from the nodes and store them within its internal data base or external cloud systems on the Internet. The frames coming from WaspMote are normally received by the 802.15.4/ZigBee radio and sent to the Internet using Ethernet, Wi-Fi and 3G interfaces.

Even though Meshlum posses several interfaces, it can not be considered a WSN. This platform is strictly thought as a gateway, so it is not a valid CWSN solution.

European Commission FP7 [78] includes two major projects involving CR technologies: CogEU [79] and CREW [80]. CogEu looks at technical, business and regulatory issues which arise in exploiting TV White Spaces across Europe, while CREW main target is to establish an open federated test platform, which facilitates experimentally-driven research on advanced spectrum sensing, cognitive radio and cognitive networking strategies in view of horizontal and vertical spectrum sharing in licensed and unlicensed bands.

The CREW platform incorporates 5 individual wireless test-beds<sup>9</sup>, as viewed at Figure 2.17, that incorporate diverse wireless technologies (heterogeneous ISM, heterogeneous licensed, cellular, wireless sensor, heterogeneous outdoor) augmented with State-of-the-art cognitive sensing platforms. Three out of these five test-beds are involved in WSN technologies: w-iLab.t, TWIST, and LOG-a-TEC test-beds. These three

<sup>8</sup><http://www.libelium.com>

<sup>9</sup>Development platforms consisting on a set of interacting devices placed at researching centers employed to obtain data and try algorithms to feedback simulators and node devices themselves.

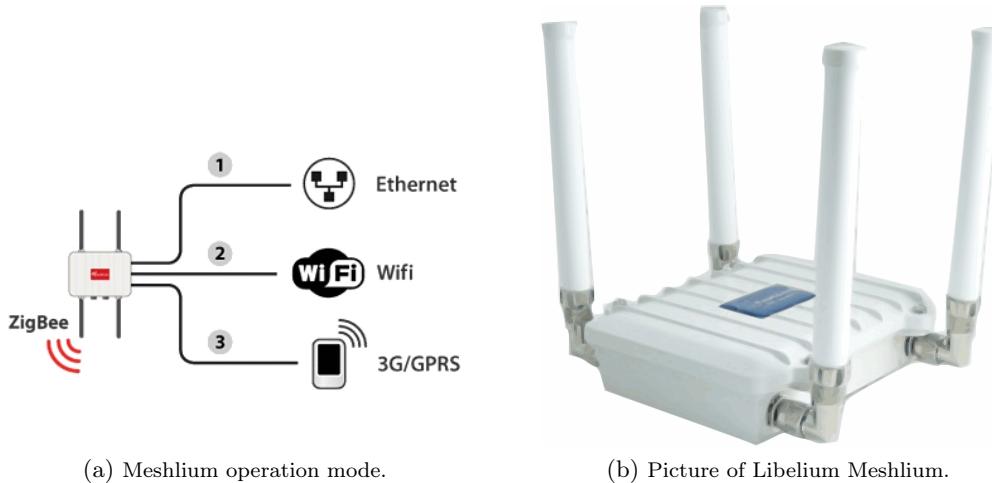


Figure 2.16: Libelium Meshlium.

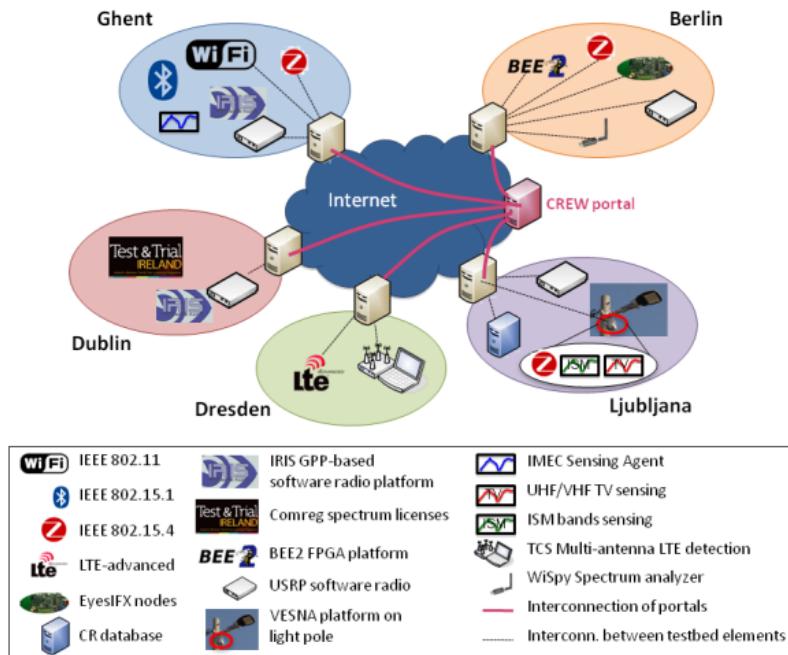


Figure 2.17: CREW Project scheme overview, obtained from [80].

platforms are explained further on together with the remaining test-bed platforms.

Main current CWSN test-beds are described below. Most of the existing, and here mentioned, test-beds do not include real WSN nodes, but emulating devices. If they host real WSN nodes, these nodes still do not meet cognitive features such as radio agility:

- *TWIST (TKN Wireless Indoor Sensor network Test-bed)* [81]. Developed by the TKN (*Telecommunication Networks Group*) at the TU (*Technische Universität*) Berlin, is one of the largest academic test-beds for experimenting with WSN applications at indoor deployment scenarios. It provides basic services like node

configuration, network-wide programming, out-of-band extraction of debug data and gathering of application data, as well as several novel features such as active power supply control of the nodes. The self-configuration capability, the use of hardware with standardized interfaces and open-source software makes the TWIST architecture scalable, affordable, and easily replicable. TWIST can be accessed locally or remotely via web interface.

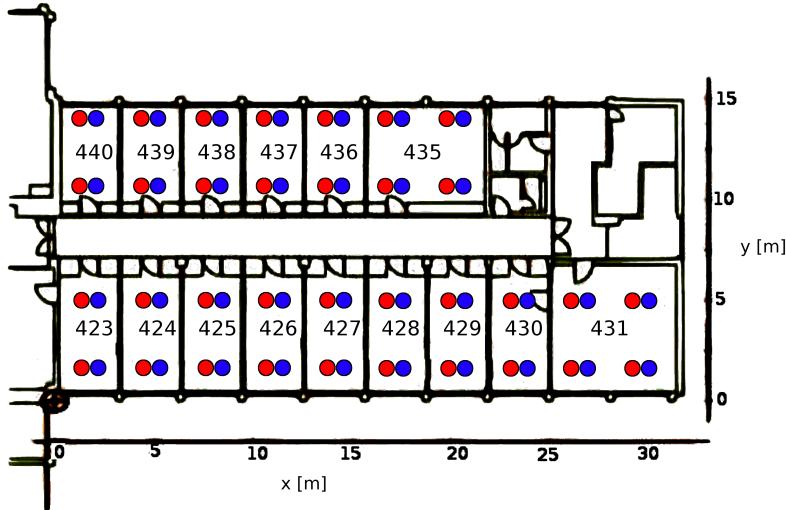


Figure 2.18: One floor deployment of TWIST test-bed.

It spans the three floors, more than 40 rooms, of an office building at the TU Berlin campus, resulting in more than 1500 m<sup>2</sup> of instrumented office space. Currently the setup is populated with 102 TmoteSky [82] nodes operating over 2.4 GHz and 102 eyesIFX [57] nodes over 868 MHz resulting in a fairly regular grid deployment pattern with intra node distance of 3 m. A set of low-cost USB WiSpy Spectrum Analyzers for the 2.4 GHz band dig over data and store it on a repository, this information is used as data-base for the CR algorithms and spectrum use optimization.

It must be clarified that nodes employed at TWIST do not posses frequency agility beyond their single frequency band. Hence, none of them can be considered CWSN nodes. Even though the test-bed supposes an approach, it is not yet a completely valid and trustable CWSN platform.

- *LOG-a-TEC* [83]. The CR experimentation part of the LOG-a-TEC test-bed is located in the municipality of Logatec, Slovenia. LOG-a-TEC is an outdoor experimental facility supporting cognitive radio networking experimentation in ISM and TV bands. It is equipped with 50 VESNA [75] platforms and VSE-ISMTV [77] boards grouped in two clusters displayed at Figure 2.19: one in the city center and one in the industrial zone. The nodes support experimentation in ISM 868 MHz (blue), ISM 2.4 GHz (red) and TV 42 - 870 MHz (green). A small mirror, consisting of 10 nodes, is also available at Jozef Stefan Institute campus. Same with TWIST tes-tbed, nodes employed here lack of frequency agility beyond their single frequency band. Hence, none of them can be considered CWSN nodes.
- *w-iLab.t*. It includes both a wireless mesh and sensor network infrastructure.

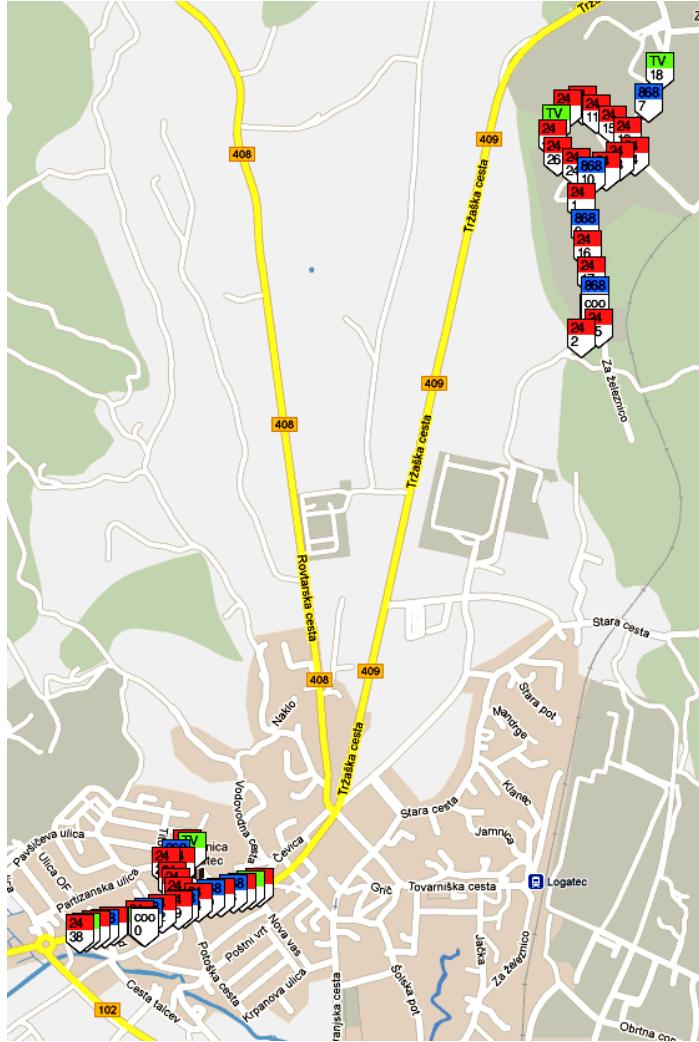


Figure 2.19: LOG-a-TEC nodes clusters deployment, obtained from [83].

These networks can interact with each other, making possible to test advanced scenarios. The test-bed is equipped with a custom designed hardware solutions, called EE (Environment Emulator). Using the EE, it is possible to emulate the behavior of any type of sensor or actuator without the need for real sensor/actuator hardware. The w-iLab.t test-bed is deployed within an office building of 18 x 90 m and it is spread out over three floors. Figure 2.20 shows some of the 200 node locations located within the iMinds [84] office premises.

Every w-iLab.t node is generic and is equipped with one or more TmoteSky [82] nodes, an iNode with 2 Wi-Fi 802.11 radios, the environment emulator and a Bluetooth interface. iNodes are Alix 3C3 [85] devices running Linux. These are mini PC's equipped with Ethernet, USB, serial, VGA (*Video Graphics Array*), audio and two IEEE 802.11 a/b/g interfaces. Finally, the EE is located in between the iNode and the TmoteSky.

Together with the previous mentioned test-beds, devices used for w-iLab do not meet CWSN features since they are either standard WSN nodes or complete mini PC modules.

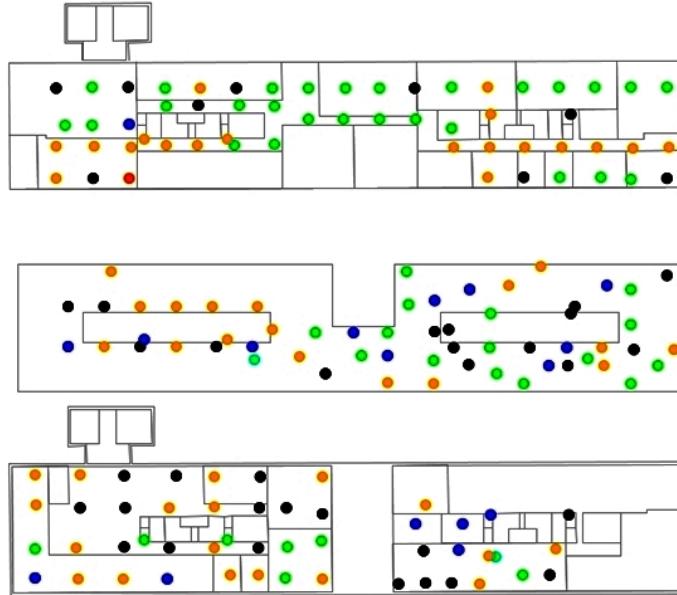


Figure 2.20: Node locations of the w-iLab.t test-bed, obtained from [80].

- *VT-CORNET (Virginia Tech COgnitive Radio NEtwork Test-bed)* [86]. It is a collection of CR nodes deployed throughout a building on the Virginia Tech main campus. The test-bed is openly available for the purpose of performing advanced CRN. The test-bed consists of a total of 48 static SDR nodes based on USRP2<sup>10</sup>, located at the ceiling throughout the ICTAS building, being placed 12 nodes per floor. In addition to the static nodes, low-power mobile nodes will also be available in order provide a research environment that accommodates a wide variety of research topics.

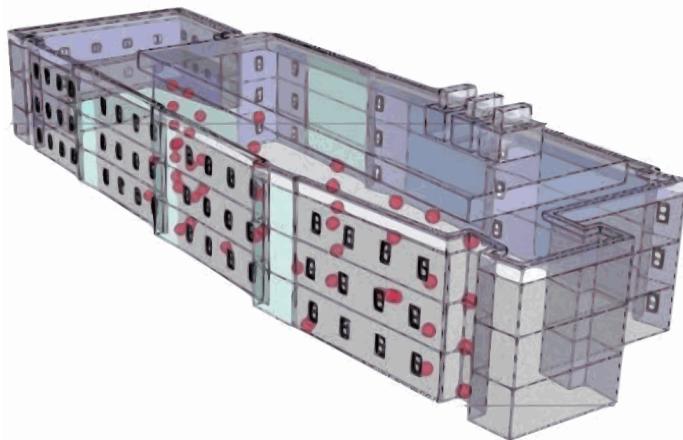


Figure 2.21: Deployment of VT-CORNET test-bed.

Remotely accessible, emphasis is on cognitive engine design, self-organizing networking algorithms, and network security. The test-bed enables researchers to

---

<sup>10</sup>The USRP family features a modular architecture with interchangeable daughter-board modules that serve as the RF front-end.

implement and test their algorithms, protocols, applications, and hardware technologies within a realistic environment.

Devices used at this platform are not real WSN nodes since their RI are based on SDR. These RIs are not suitable for WSN regarding their high power consumption. Despite their possibilities for frequency mobility, the solution implemented by this test-bed is not a real CWSN implementation.

- *FIT/CorteXlab - Cognitive Radio Test-bed* [87]. This test-bed, still under deployment, will be hosted at INSA-Lyon, in France. CorteXlab will suppose one more researching center of the Future Internet of Things project [88]. CorteXlab will use the network architecture developed in SensLAB [89] and will integrate SDR nodes to offer a remotely accessible development platform for distributed CR. Reconfigurability, compatibility, coexistence and even cooperation between SDR nodes will be evaluable. A large set of heterogeneous SDR nodes (MIMO (*Multiple-Input Multiple-Output*) nodes, SISO (*Single-Input Single-Output*) nodes and WSN nodes) together with classical sensor nodes will permit a full experimental evaluation.



Figure 2.22: CorteXlab test-bed logo.

The test-bed will be installed in 180 m<sup>2</sup> shielded room (isolated from any external interference) and also partly covered with EM (*Electro-Magnetic*) absorbing material. Depending on the set of enabled frequencies, the design of the room will enable to control the radio channel characteristics (number of paths, delays, etc.) and to ensure a high level of reproducibility of experimentations.

Nodes will be uniformly distributed. These nodes will be able to accept PHY layer implementations on both hardware, i.e. FPGA, and software on general purpose CPUs. Furthermore, they will be capable of outputting performance metrics, such as throughput, BER (*Bit-Error Rate*) and power consumption. The nodes are interconnected through a high speed Ethernet link, in order to allow for cooperation and sharing of information. A unified server will also be available for starting, coordinating and collecting the results of experimentations. Experimentations themselves can be from PHY layer up, including the possibility of cross-layer interactions.

Despite the large amount of devices employed at this test-bed platform that combines different architectures and nodes, none of them are real CWSN devices. Frequency mobility is achieved over SDR modules, which can not even be considered for WSN implementations. SDR devices gather features not WSN-compatible that collide with low-resources and low-consumption natures.

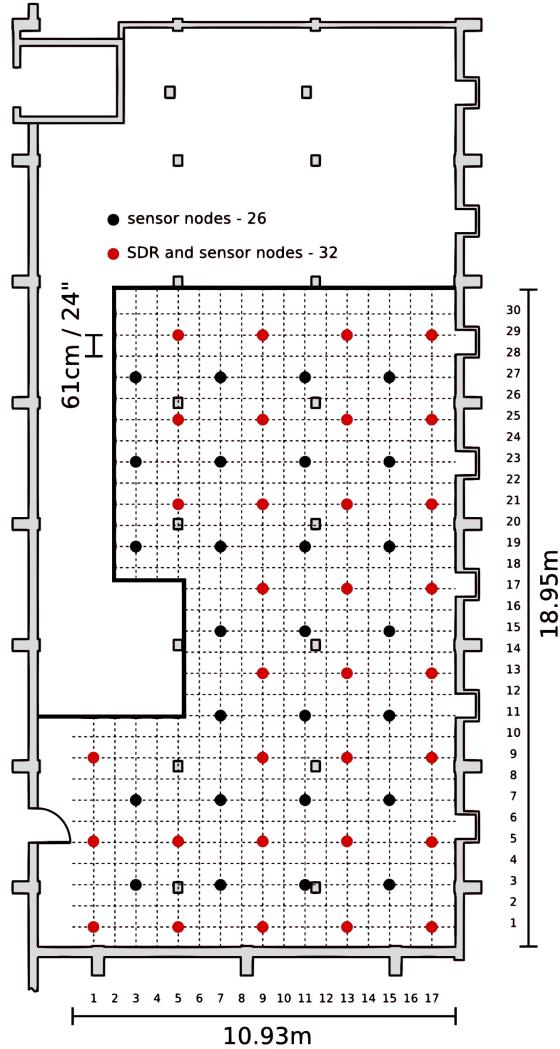


Figure 2.23: CorteXlab test-bed deployment, obtained from [87].

#### 2.4.1.3. Standards

Standardization is at the core of the current and future success of cognitive radio. The IEEE 802.22 [24] working group is developing what will be the first cognitive radio-based international standard for WRAN (*Wireless Regional Area Network*) with tangible frequency bands for its operation. It will operate over unused television channels. One system such as Unlicensed-Band Cognitive Radio, which can only utilize unlicensed parts of the RF spectrum, is described in the IEEE 802.15 Task Group 2 [19] specifications. This standard focus on the coexistence of IEEE 802.11 and Bluetooth. Many other standards such as Wi-Fi (IEEE 802.11), Zigbee (IEEE 802.15.4), and WiMAX (IEEE 802.16) already include some degree of CR technology today.

## 2.5. Contribution

It has shown how CWSN is still an immature researching field where scarce devices are found, and all of them incomplete. Real and complete devices to allow develo-

pers to try security, energy, QoS, and spectrum optimization through new algorithms and strategies, are needed. Facing this need, the LSI proposes the implementation of a complete development platform, hardware and software, fully oriented to CWSN investigation, able to truly implement real WSN applications. This goal defines and frames this Master Thesis.

# Chapter 3

## Review Study

*Work it harder, make it better,  
do it faster, makes us stronger,  
more than ever hour after,  
our work is never over*

Thomas & Guy-Manuel, Daft Punk

A model for our cognitive node is defined in this chapter. Initially, the First Cognitive Device is evaluated to obtain some guidelines and potential improvements facing our design. The final device implementation requirements and constraints are defined. Finally, a discussion and evaluation over different modules is carried out, exposing different commercial options and some final conclusions.

### 3.1. Cognitive Wireless Sensor Network Node Model

Basically, the CWSN node scheme responds to a standard WSN model based on a microcontroller, radio and sensor interfaces, and power supply. However, a further review shows some general differences:

- *Multiple Radio Interfacing*: Regarding CR capabilities for spectrum management and DSA, the node must include multiple RIs enabling different spectrum bands instead of a single one. This capability is referred to the hardware modules.
- *Multiple Control/Sensor Interfacing*: Since the device will serve as a standard CWSN platform, it must embed different multi-purpose generic interfaces<sup>1</sup>. This fact affects the hardware design of the device.
- *Protocol Stacking*: In order to enable communication over the different interfaces, the model must provide proper access to the lower layers of a communication protocol stack for each RI<sup>2</sup>. This affects the running firmware.

---

<sup>1</sup>These interfaces cover the usual peripherals and buses such as I<sup>2</sup>C, SPI, UART, USB, etc.

<sup>2</sup>As it will be described, the employed firmware supposes a great software development since it maintains a single stack shared by all the interfaces.

- *Cognitive Layer*: Software implementations responsible to carry out all the cognition algorithms and computations. KP and data flow management. Extra compute capabilities might be needed to give support to this layer. The already named CRmodule must carry out functions belonging to this domain.

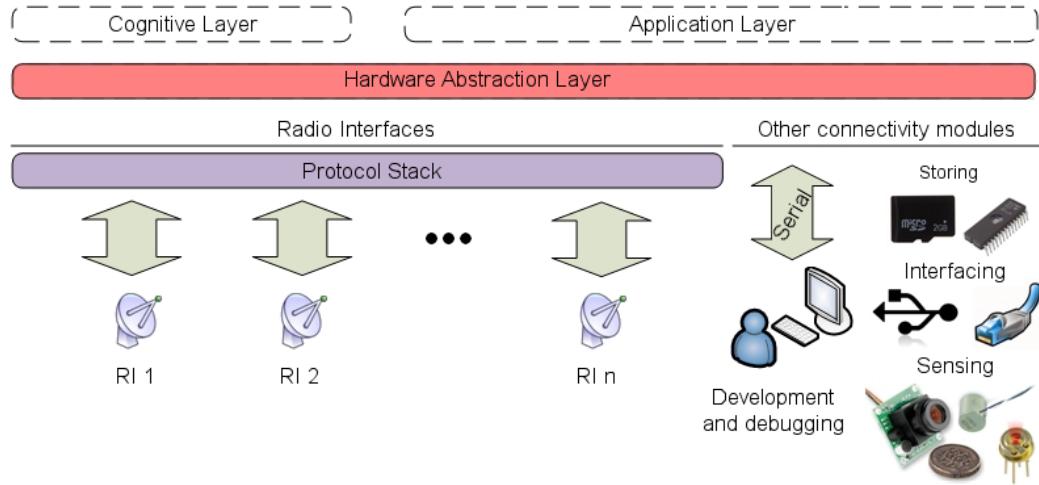


Figure 3.1: CWSN node model.

Figure 3.1 gives a view about the basic sought model. It shows together software and hardware modules in a simplified way. Additionally, cognition tasks compulsory require several functions to take into account at the design stage:

- *Power management*. Efficiency and low-power consumption are crucial following the inherent trend in WSNs. Once deployed, it is common not having or barely having access to the nodes. Replacing batteries to an entire WSN might become an infeasible task. The model should be able to include an autonomous power supply and other functions:
  - TX and RX (*Reception*) power regulation.
  - Unused modules disconnection.
  - Power-saving sleeping modes for transceivers and MCU.
- *Spectrum management*. Some of the mainly needed functions:
  - RSSI (*Received Signal Strength Indication*) detection.
  - Channel switching.
  - Energy scans all over the available frequencies range.

All the previous considerations give support to establish cognitive capabilities to the final application. These considerations, along with the requirements to be defined, will shape the model of the final cNGD implementation.

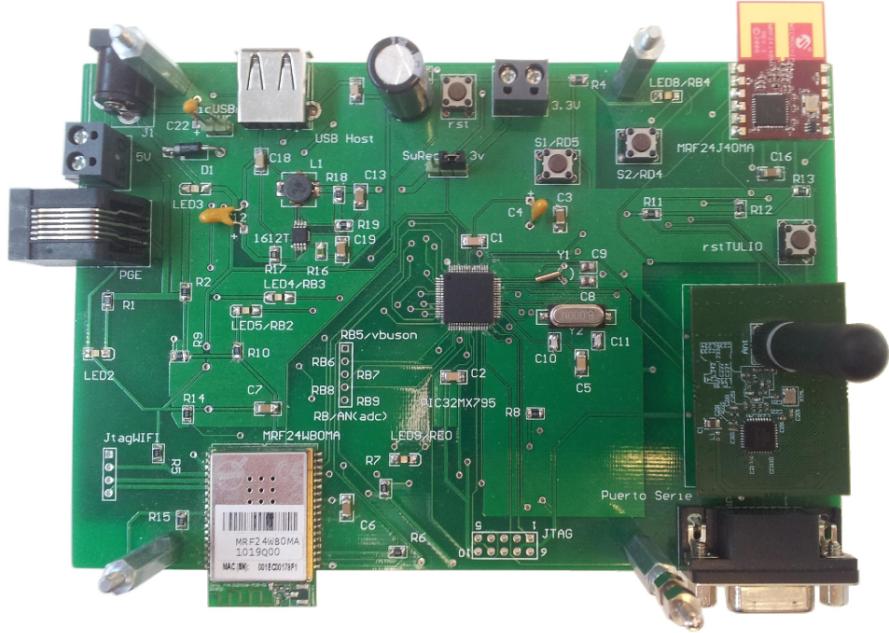


Figure 3.2: Picture of the FCD, developed at LSI in 2011, obtained from [9].

### 3.2. First Cognitive Device review

The FCD was developed in 2011 at the LSI by Fernando Lopez Lara [74]. It supposed the first hardware platform fully oriented to CWSN development that included three RIIs. Integrating hardware and software, it gave the chance to develop and test algorithms, strategies and applications for CWSN. Moreover, it allowed to analyze suitability for radio interfaces, compute capability, and other components. It gave the strengths and weaknesses to establish the fundamentals for future designs.

While constituting the first device meeting its features, the FCD introduced several improvable issues found after its implementation and evaluation. Its power consumption, size and cost, together with other not achieved but desired requirement, make the device unsuitable for CWSN test-bedding purposes and claim for a new design [9].

The FCD has a Microchip PIC (*Peripheral Interface Controller*) hosting a MIPS (*Microprocessor without Interlocked Pipeline Stages*) M4K, a 32-bits MCU unit. It has a Harvard architecture employing a five-stages pipeline and a RISC (*Reduced Instruction Set Computer*). The chosen model, a PIC32MX795F512H [90], integrates a 512 KB Flash Memory and a 128 KB general purpose SRAM (*Static Random Access Memory*). It allows a maximum 80 MHz clock frequency and it achieves a throughput as high as 1.65 DMIPS/MHz<sup>3</sup>. This PIC was found to offer too high features for our requirements, hence existing a reducible amount of power consumption on this module. Reducing capacity with regards to Flash and RAM (*Random Access Memory*) memory is seen as a positive option.

The FCD includes three RIIs enabling access to two different spectrum regions. Two

---

<sup>3</sup>DMIPS (*Dhrystone Million Instructions per Second*) divided by CPU (*Central Processing Unit*) frequency. Dhrystone is a synthetic computing benchmark program become representative of general processor performance.

of the selected interfaces work on the 2.4 GHz band and the third one over the 868 MHz, corresponding to the ISM bands in most part of Region 1. To operate over the 868 MHz band it used a Tulio<sup>TM</sup> [91] transceiver oriented to LoWPAN (*Low-Rate Wireless Personal Area Network*) and IEEE 802.15.4 compatible. For the 2.4 GHz band, two Microchip modules were selected. A MRF24J40 [92] module using Microchip proprietary protocols such us MiWi<sup>TM</sup> P2P [93], MiWi<sup>TM</sup> Mesh, or MiWiPRO<sup>TM</sup> [94] and a MRF24WB0MA [95] module, IEEE 802.11 compatible, as Wi-Fi. Firmware running on the FCD operated by using three different protocol stacks. Tulio<sup>TM</sup> firmware was provided by AWD and communications with the MCU take place through an UART. Stacks for MiWi [47] and TCP/IP [96] protocols are provided freely by Microchip, being the first one significantly lighter and simpler than the second one. Communications with these transceivers use SPI interfaces.

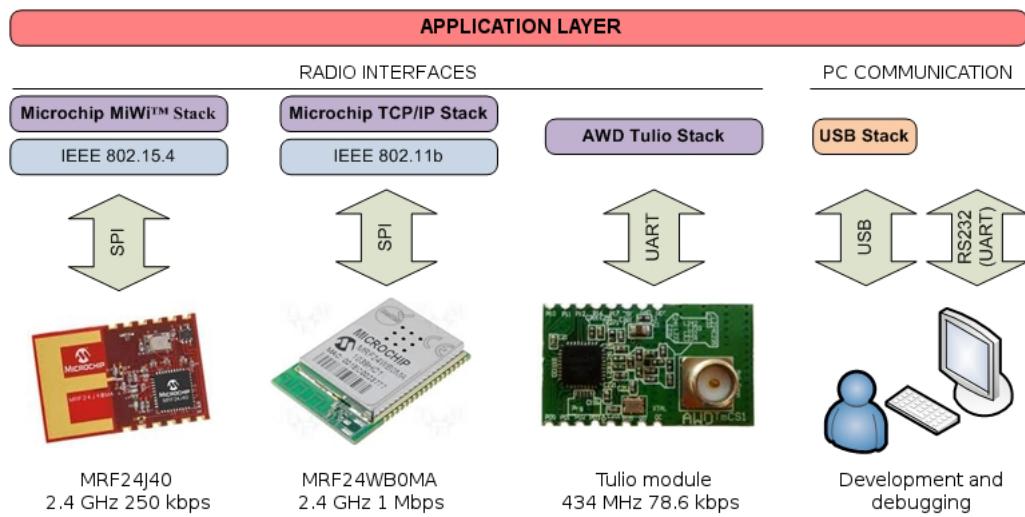


Figure 3.3: Architecture model of the FCD, developed at LSI in 2011, obtained from [9].

The Tulio<sup>TM</sup> module offers a very low-consumption but it contains a Texas Instruments microcontroller increasing complexity on the global structure, moreover, this fact raises the global consumption up. Maintaining this interface will be disputed. On the other hand, the Wi-Fi interface seems to be inappropriate for WSN due to its high power consumption, hibernation-related problems, and its complex, resources-consuming communication stack. Even though this interface is thought as a gateway, new options are highly needed. MRF24J40 gives a throughput suitable for the already described requirements, thus it could be a good candidate for the new design.

Regarding firmware options, the situation resulting from combining three different stacks becomes messy, confusing, power and resources-consuming, not suitable to deal properly with an application and a cognitive layer. Hence, using the developed firmware and HAL [9] that integrates three stacks and presents a straightforward API (*Application Programming Interface*) for upper layers, seems a good choice for this work.

The node comprises two USB terminals and an RS232 interface for debugging and interconnection purposes. Power supply system accepts 3.3 V and 5 V options including a voltage regulator and a charge pump.

Maintaining all these interfaces supposes larger device size and greater power con-

sumption. A review about keeping all them as well as all the power supply options will be brought to pass, some of them seemed useless and inefficient.

Regarding other issues, reducing the general size, making the device more scalable or modular are indeed needed on the cNGD.

### 3.3. Firmware review

The work in [9] described an implemented new firmware for the cNGD. The firmware runs over PIC32 architectures. It was developed over the FCD but slightly adapting the hardware for better performance [10]. The firmware offers access to three different regions over the radio spectrum by means of Microchip transceivers and an appropriately modified and adapted MiWi<sup>TM</sup> protocol stack that allows using more than one PHY layer.

In order to abstract the application from the hardware and software complexity design, the firmware includes a straightforward HAL to deal easily with the RIs within a close set of functions. In this way, node usability improves and applications cost and developing time decrease.

The new firmware supposes several advantages:

- Versatility and adaptability are given. Hardware modifications and new modules are possible with minimal software adaptation.
- Firmware takes care of the protocol stack management.
- MiWi<sup>TM</sup> protocol stack adaptation makes a better efficient use of resources and removes complexity.
- Devices fabrication costs are reduced since some modules are not needed anymore and less featured controllers are suitable.
- Sleeping modes provided by the firmware bring the key for autonomous operation modes. Whenever circumstances permit it, enabling sleeping modes will drop significantly the power consumption.

This software module must be taken into account at the design stage for its inclusion into the cNGD. It also might be needed of adaptation or debugging tasks due to its short life-time and usage.

### 3.4. Cognitive Radio Module review

This module is designed to operate under PIC32 architectures. It was thought to run over the previous described firmware. Its development took place over the FCD and all the implementation process is set out on [10].

This software module represents an actual realization of the architecture described in [11] and, somewhat, enhances it by introducing slight changes.

An overview to the general performance shows an scheme based on different modules such as Repository, Discovery, Optimization, Execution, Access Control, Policy, Messenger and Virtual Control Channel. Each module performs a defined tasks and connect each other attending to defined policies. Figure 3.4 illustrates the general scheme, for a further view consult the given references.

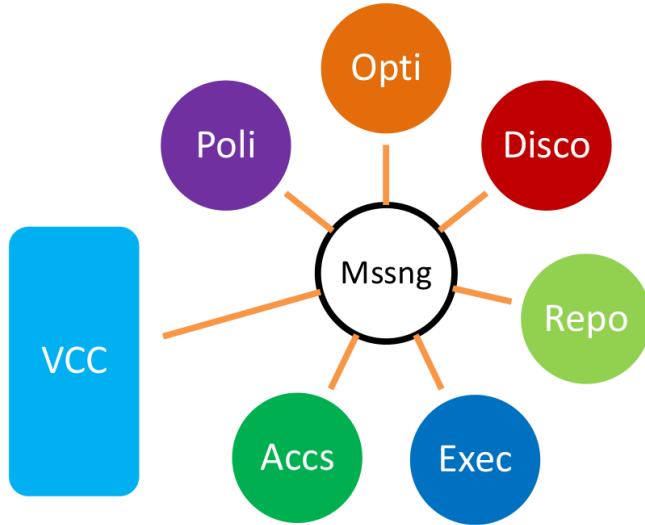


Figure 3.4: CRmodule architecture, obtained from [10].

### 3.5. System Requirements

Once described the CWSN node model and after evaluating the main related developments, it is time to define the requirements for the cNGD. These are classified as follows:

- Essential requirements
  - Communication possibilities, at least, over two ISM bands (868 MHz and 2.4 GHz). Having, as far as possible, fully-configurable transceivers.
  - Modularity.
  - IEEE 802.11 standard inter-operation possibilities.
  - External pluggable antenna possibilities.
  - Development-oriented. Comprising debugging tools.
  - Working under a single development framework.
  - Useful and powered to try concepts referred to optimization strategies for:
    - Security.
    - Power consumption.
    - Synchronization.
    - QoS.
- Desirable features
  - Portable and autonomous power supply options.
  - Remote application-loader.
  - Reduced size.
  - Communication over three ISM bands (433 MHz, 868 MHz, 2.4 GHz).
  - Mono-core-based architecture.

- Battery charge-state monitoring.
- Real radio headers switching possibilities so as to use a single transceiver for more than one frequency band.

In addition, it should respond to standard requirements for WSNs:

- *Scalability*. Given the potentially large amount of nodes composing a WSN, the model claims for a easily scalable design.
- *Cost*. Due to the same previous reason, low-cost is a requirement. Otherwise their price make networks unrealizable.
- *Ubiquitous computing*. Each node hosts some computing capacity, this gives to the network the chance for distributed cognition. Hence, robustness and stability is given to the set.
- *Power consumption*. It must be prepared to operate supplied by a battery and achieve a certain autonomy.
- *Throughput*. Performance must be high enough for a test-bed platform. Regarding the generic nature of the platform, it should be prepared for a wide range of applications. Cognition, on the other hand, increases as well the need for computation capabilities.
- *Communication data-rate*. Not too high data-rate transmissions are usually needed. Few tens or hundreds of kbps are enough.
- *Security*. Radio Interfaces allowing manageable security mechanisms. Being possible to safely deal with confidential or private data.

## 3.6. Technology

Step by step, all the needed technologies for our design will be evaluated and compared to the employed by the FCD.

### 3.6.1. Microcontroller

Based on the requirements described in 3.5, it seemed necessary to keep the number of cores reduced to one unit. Furthermore, using a 32-bits Microchip PIC was an imposed design decision, as it was said.

The criteria to choose a microcontroller were:

- The less featured and simplest microcontroller that suits our operation memory parameters. FCD showed that a Flash memory above 200 KB was enough and a RAM memory over 32 KB fits our needs. This helps to keep down the microcontroller power consumption. All 32-bits PICs family posses a high enough throughput.
- Some peripherals are essential to accomplish required tasks. At least three SPIs<sup>4</sup> would be needed to interface the microcontroller and the radio transceivers. In

---

<sup>4</sup>Most of commercial radio transceivers set their communication with the MCU over SPIs

addition, peripherals for debugging such as UARTs or USBs must be simultaneously available. Apart from that, it should contain other buses and accessible peripherals for modularity. Probably a 100-pins microcontroller could provide better access to multiple peripherals than smaller sizes.

- Lowest price.

The different options included at Microchip catalog, described and showing their main features, are shown in table 3.1.

Microcontroller	Max. Frequency	32-bits Microchip microcontrollers										ADC	Pins	Consumption	Price (€)			
		Memory (KB)		Peripherals														
		Flash	RAM	SPI	UART	I2C	USB	CAN	ETH									
PIC32MX664F128H	80	128	32	3	4	4	0	1	Full Speed Host Dev. OTG	10/100 BaseT	16	64	Run: 39 mA	5.80				
PIC32MX664F128L				4		5						100	Idle: 17 mA	6.41				
PIC32MX764F128H			32	3		4						64	Sleep: 20 µA	5.95				
PIC32MX764F128L				4		5						100			6.54			
PIC32MX675F256H		256	64	3	4	4	0					64	Run: 85 mA	7.11				
PIC32MX675F256L				4		5						100			8.35			
PIC32MX775F256H			64	3		4	2					64	Idle: 36 mA	8.07				
PIC32MX775F256L				4		5						100			8.79			
PIC32MX775F512H		512	128	3	4	4	2					64	Sleep: 41 µA	8.75				
PIC32MX775F512L				4		5						100			9.47			
PIC32MX795F256H												64	Not Specified	9.47				
PIC32MX795F256L												100		9.57				

Table 3.1: Comparative table of 32-bits family Microchip PICs.

### 3.6.2. Radio Interfaces

One of the desired requirements is enabling radio activity over three ISM bands (434 MHz, 868 MHz and 2.4 GHz), so in order to achieve it, compatible transceivers on these bands must be studied. We first took two decisions based on the experience brought by the FCD.

The FCD access the 868 MHz band using a Tulio<sup>TM</sup> module. This module brings global complexity to the design, it includes a core itself and the communication between cores are driven by the UART. To keep the single-core architecture, lowering-down power consumption and meeting the described requirements, this module will be suppressed and it will be looked for a replacement.

MRF24WB0MA Wi-Fi transceiver, embedded by the FCD, raises up the power consumption to unsuitable levels in a WSN. The protocol stack is very computationally heavy and memory-consuming compared other usual protocol stacks in WSNs. New ways to establish a IEEE 802.11 gateway will be studied, maybe through a pluggable expansion module. The low-consumption, low-resources and simple scheme sought in the cNGD cannot afford to include this transceiver.

Table 3.2 shows transceivers and modules supported by the chosen firmware that enables communication at the desired bands. Modules such as the MRF49XA PICtail<sup>TM</sup> Daughter Board, Figure 3.5, whose size does not suit our requirements, have not been considered.

There are not complete modules available for 434 MHz band. For 868 MHz, module MRF89XAM8A is available. However, MRF49XA transceiver, which provides communications at these two bands, is susceptible to be included into an ad-hoc designed RI board. Figure 3.6 gives a comparison among wireless networking protocols.

Model	Freq. (MHz)	Compliant Technologies	Data Rate (kbps)	Sensing options	Consumption				Price (€)
					Tx	Rx	Sleep	Idle	
MRF49XA (transceiver)	434/ 868/915	MiWi™	250	RSSI	I433=26 mA @ 7 dBm I868=27 mA @ 5 dBm	I433=13 mA. I868=14 mA	0.3 μA	1.2 mA	2.84
MRF89XA (transceiver)	868/ 915/955		200		25 mA @ +10 dBm	3.5 mA	0.1 μA	60 μA	3.86
MRF89XAM8A (module)	868		FSK: 40 OOK: 16		16 mA @ +1 dBm				8.26

Table 3.2: Comparative table of sub-GHz Microchip solutions

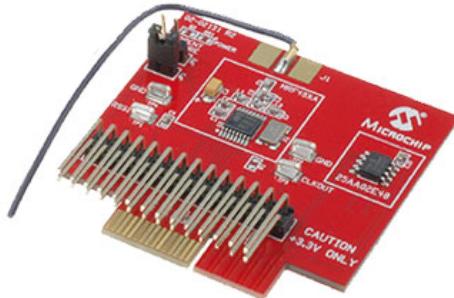


Figure 3.5: MRF49XA PICtail™ Daughter Board

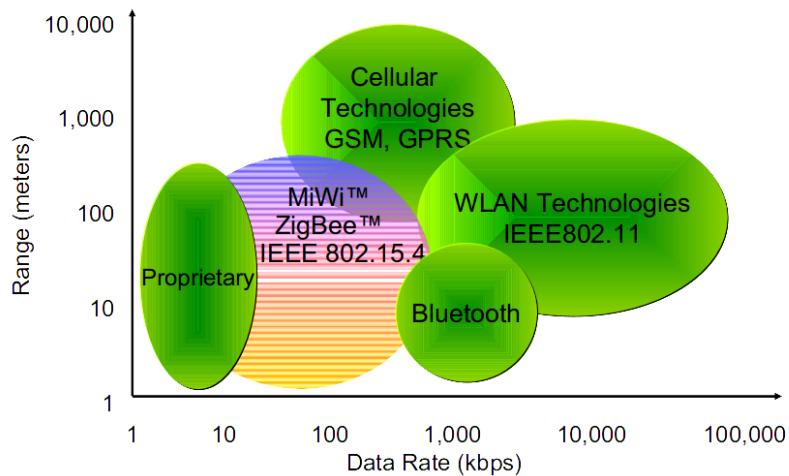


Figure 3.6: Main wireless networking options comparison

### 3.6.3. Serial Communication

Serial communication is a very important feature for applications development and debugging. It allows wiring the node and establishing communication between the cNGD and any other device, normally a computer or storing devices to track data. Peripherals such as the UART is commonly used to establish these communication tasks. More recent implementations enable communications through USB.

While the FCD includes two USB connectors, a  $\mu$ USB and an A-type, together with an RS232 communication module, three communication ways to interface the node seems too many. These modules are size and power consuming.

$\mu$ USB connectors show a reduced size compared to first connectors and they do

expand the first connectors capabilities.

The idea of including an A-type USB connector gives some interconnection opportunities with attachable devices like mouses, keyboards, or portable memories in which the node adopts a *master* role facing a *slave* device. This feature supposes the inclusion of a power consuming circuitry to raise the voltage up to 5 V into the USB, subtracting simplicity to the design. In order to fit the requirements, it does not seem to be needed this last capability on each node. A  $\mu$ USB connector would be enough.

RS232 communication so does need its own power supply. The size and cost for an RS232 connector and circuitry, which requires a MAX3232 chip, is too high for a default serial communication module within each node. An optional module for RS232 might be a good option given its wide acceptance and utility in embedded systems despite the global use of USB.

### 3.6.4. Power Supply System

The power supply system designed for the FCD includes some modules susceptible to be suppressed or replaced to keep the simplicity and reduce the consumption.

The MCP1612 [97] buck regulator specifications fit the supply requirements on excess, so a less-featured component might be enough at the new design. The maximum output current for the MCP1612, up to 1 A, could be significantly reduced.

The MCP1253-33X [98] charge-pump, responsible for raising the voltage up to 5V to operate over USB as master, could be removed in the new design if this option is not finally implemented.

### 3.6.5. Timing

Timing options included on the FCD suppose a low-consumption oscillator for real-timing and a second oscillator for maximum operation frequency. It does not appear any problem related to the timing configuration. The defined configuration fits the requirements and, firstly, seems a good option to keep it unchanged.

### 3.6.6. Expansion Capabilities

Application capabilities over the FCD are poor. Apart from interacting with the RI and serial communication, the board does not offer options for sensing/controlling applications neither other more complex possibilities such as including new communication modules. This is because the FCD does not embed any expansion header or slot where adapting any new design to expand to new features.

The new design calls for expansion options where application capabilities could be widely and easily opened by attaching modules including an ad-hoc design. A complete expansion header for a WSN node should include access to the following peripherals/buses:

- *I<sup>2</sup>C*. Being the most common bus to interface sensors to the MCU.
- *USB*. For serial communication, mainly with a host device.
- *UART*. For asynchronous serial communication with devices.
- *SPI*. As a usual serial communication peripheral.

- *Ethernet.* For IEEE 802.11 compatibility.
- *Interruptions.* Enabling interrupting the MCU without employing polling.

### 3.7. Conclusions

After having a review all over the, hardware and software, recent implementations related to CWSNs that might bring some guidances in our design; once evaluated the used technology and analyzed the weaknesses; having defined some minimal requirements to fit, and being them applied to our evaluations, the obtained conclusions to face the design stage follow:

- The model implemented by the FCD is quite close to the one pursued. It will serve as guide in our designing process since its review reveals significant technology and behavioral-related information.
- A new configuration regarding size and cost that allows flexibility and modularity is essential.
- Implemented software such as the described firmware will help us saving time on the software development and it proposes a very complete solution that fits our requirements. It provides a shared protocol stack and a very useful HAL. Even if needed some slight adaptations, it must be compatible with our cNGD design.
- FCD includes a too featured microcontroller for our requirements, implying higher power consumption and cost. A new MCU seems needed.
- RIs used by the FCD open communication on two frequency bands. They show a very inefficient performance and high consumption behavior due to:
  - Coexistence of three heterogeneous transceivers and different protocol stacks. Bringing complexity and operation inefficiency.
  - Presence of a Wi-Fi transceiver, whose consumptions goes up to 160mA at TX.
  - A TULIO<sup>TM</sup> transceiver whose inclusion, hosting a core by itself, supposes a multi-core implementation.

A reconfiguration of the RIs is needed. Bringing homogeneity and a WSN standard protocol such as IEEE 802.15.4. In addition, desirable requirements call for a third frequency band to implement. The lack of suitable modules over 434 MHz band claims for a custom RI design.

- Serial communication system claims for simplifications, the same as the power supply system.
- Expansion capabilities enhance modularity and node versatility. Expansion pluggable options must access the main peripherals. Enabling the 802.11 standard is a requirement.
- A portable supply system must urgently be completely developed.



# Chapter 4

## Design Study

*Surf it, scroll it, pause it, click it,  
cross it, crack it, switch - update it  
name it, rate it, tune it, print it,  
scan it, send it, fax - rename it  
touch it, bring it, pay it, watch it*

Thomas & Guy-Manuel, Daft Punk

In this chapter, the platform design is described. The design decisions, specifications, schemes and needed calculations will be exposed. The detailed characteristics of any chosen component, (including microcontroller, radio interfaces, and power-supply options) will be provided.

### 4.1. Global Hardware Description

Attending to the already described requirements and the conclusions obtained from the FCD review, the following decisions have been taken before proceeding with the design process.

The whole system is thought as a teaser consisting of different nestable modules that give flexibility to the scheme. The main module, cNGD, is the cornerstone. cNGD composes the main platform of the design. It hosts the power supply system, provides the possibility to embed up to three RIs, posses a control core unit, and offers other interfacing possibilities that are discussed later in this thesis. As part of this modular design, the interfaces that cNGD hosts open possibilities to new extension board implementations through a pair of headers. Following popular terminology, these attachable boards will be referred to as shields.

Offering the chance of embedding up to three RIs, the device could access three different frequency bands over the spectrum, and fulfills one of the desired requirements described in Section 3.5. The chosen operation frequencies in our design are 434 MHz, 868 MHz, and 2.4 GHz, coupling the ISM bands in most part of Europe. These bands are preferred for WSN due to the data-rate, transmission power, and transmission range parameters they offer for no cost. This configuration places the device in the bounds of complexity and cost. At the same time, this feature is essential for a CWSN proper investigation, giving radio communication opportunities not provided by any

other similar device so far. Moreover, a full exploitation of the firmware described in 3.3 can be done, since it offers an integrated MiWi<sup>TM</sup> stack for three RIs.

The three RI are based on the IEEE 802.15.4 standard for WPANs, commonly used in WSN. Specifically, the set of interfaces will operate under the MiWi<sup>TM</sup> or MiWi<sup>TM</sup>P2P protocol. They both are proprietary protocols designed by Microchip Technology that uses small, low-power digital radios. They are an open source option designed for low data transmission rates (up to 250 Kbit/s) and short distance (up to 100 without obstacles), cost constrained networks. The main difference between MiWi<sup>TM</sup> and ZigBee<sup>TM</sup><sup>1</sup> is complexity. MiWi<sup>TM</sup> offers a much more simple operation, resulting in a lighter implementation. The size required for the MiWi<sup>TM</sup> stack is 3-10 KB on the ROM (depending on the node role) while ZigBee<sup>TM</sup> takes 20-40 KB. MiWi<sup>TM</sup> is free of charge and it does not require a license acquisition as long as Microchip components are used. In fact, Microchip requires the use of MiWi<sup>TM</sup> for its products. The global design becomes simpler and more homogeneous, sharing a common communication protocol for all the RIs.

Since compatible and suitable size commercial options for the 434 MHz band were not found, a custom full implementation of this interface was required. This implementation is based on the MRF49XA transceiver, which can operate also at 868 MHz. In order to provide homogeneity, (and since it does not suppose a great extra effort), the new RI design will be used for both 434 and 868 MHz bands. These RI will be called  $\mu$ Trans 434/868 from now on. Full description of  $\mu$ Trans 434/868 is given in Section 4.2. The option at 2.4 GHz is a MRF24J40MA module.

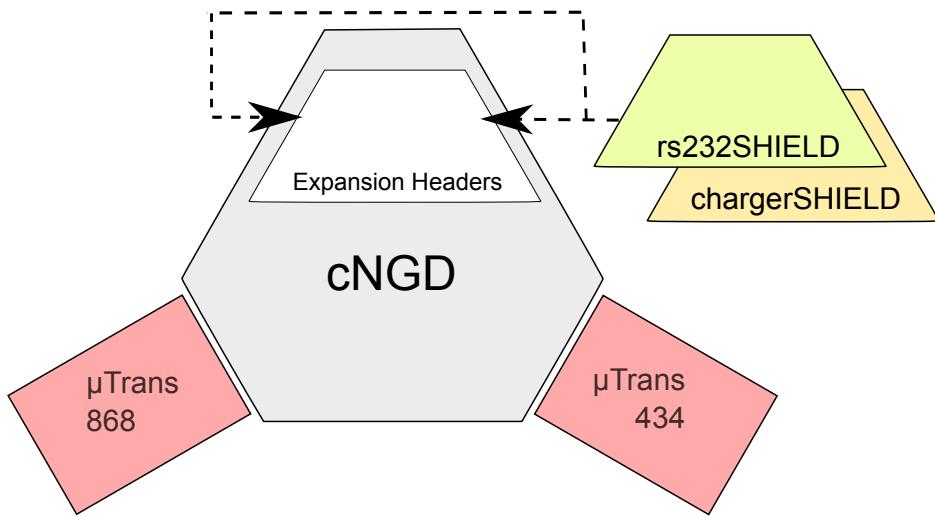


Figure 4.1: Global hardware modules of the platform to develop.

For this work, a pair of shields was designed. A simple serial communication complement links one UART and a classic RS232 interface over the so-called rs232SHIELD. This interface would facilitate the software development since the firmware just allowed debugging through the UART at first. It is described in Section 4.4. A Ni-MH (*Nickel-Metal Hydride*) and Ni-Cd (*Nickel-Cadmium*) battery charger was also designed and implemented as a utility for portability options. The device was called chargerSHIELD.

Therefore, the set of designs to carry out, showed at Figure 4.1, includes  $\mu$ Trans

<sup>1</sup>A very popular wireless IEEE 802.15.4 compatible protocol employed at WSNs.

434 and 868, cNGD, rs232SHIELD and chargerSHIELD. All hardware design files can be accessed freely at the public GitHub<sup>2</sup> repository at <https://github.com/agus-xyz/cNGD-hard>.

## 4.2. Transceivers - $\mu$ Trans 434/868

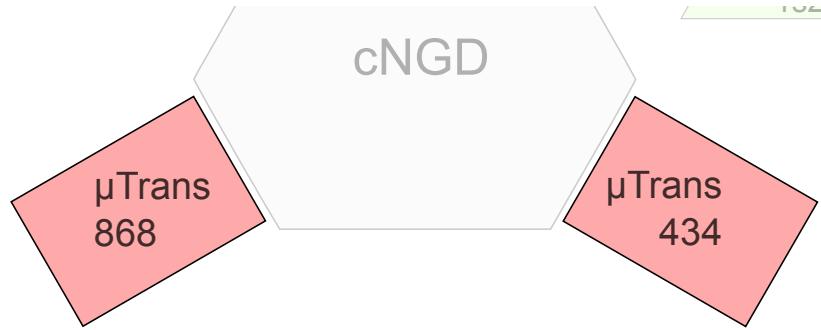


Figure 4.2:  $\mu$ Trans detail over global hardware modules to develop.

The chosen firmware for our platform gives support for MRF49XA modules over 434 MHz and 868 MHz bands. Specifically, the modules used during the firmware development were the MRF49XA PICtail<sup>TM</sup> Daughter Board [99]. This board is a demonstration and development daughter board for the MRF49XA ISM band sub-GHz RF transceiver. The board can be plugged into a suitable header, which makes it unsuitable for a reduced size and robust WSN design. It either accepts connection to external antennas, which is a desirable requirement.

Previous to the cNGD designing, which will embeds the  $\mu$ Trans 434/868, the design of these RIs was needed.

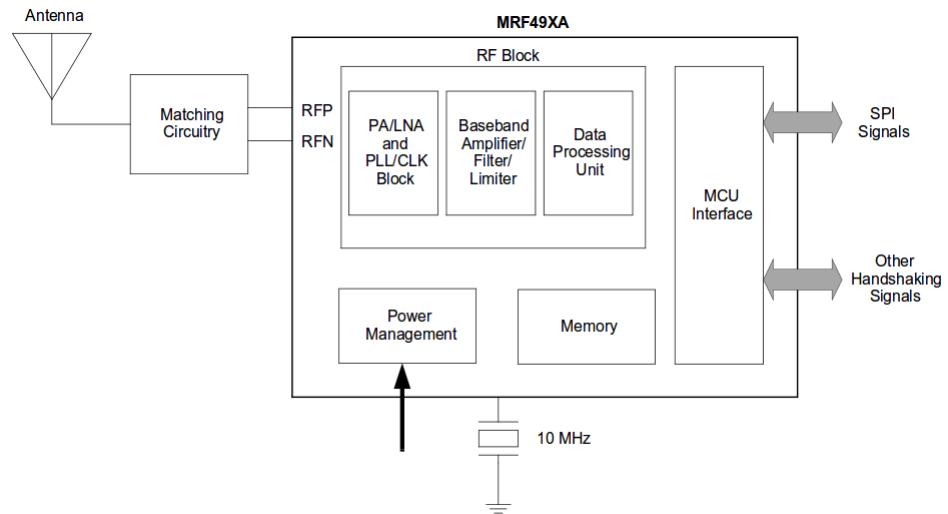


Figure 4.3:  $\mu$ Trans 434/868 blocks diagram, obtained from [100].

<sup>2</sup>Web-based hosting service for software development projects that use the Git revision control system

#### 4.2.1. Description

MRF49XA is a low-power fully integrated Sub-GHz RF transceiver. An ideal choice for low-cost, high-volume, low data rate (<256 Kbps), two-way, short range wireless applications. It can operate in the unlicensed 434, 868 and 915 MHz frequency bands. The transceiver is integrated with different sleep modes and an internal wake-up timer to reduce the overall current consumption. The device operates in the low-voltage range of 2.2 to 3.8 V, and in Sleep mode, it operates at a very low-current state, typically 0.3  $\mu$ A.

Further details are given below:

- Modulation Technique: FSK (*Frequency-Shift Keying*) with FHSS (*Frequency Hopping Spread Spectrum*) capability.
- Differential RF Input/Output:
  - -110 dBm Typical sensitivity with 0 dBm maximum input level.
  - +7 dBm Typical transmit output power.
- Maximum supported data rates:
  - Digital mode 115.2 Kbps.
  - Analog mode 256 Kbps.
- Current Consumption:
  - TX. 434MHz, I = 23 mA at 7 dBm. 868MHz, I = 24 mA at 5 dBm.
  - RX. 433MHz, I = 11 mA. 868MHz, I = 12 mA.
  - Idle. 1.2 mA (Max).
- Temperature range of operation from -40° to 85°.
- Other functionalities: DQI (*Data Quality Indicator*) data, RSSI data.

This transceiver is not IEEE 802.15.4 compatible since the number of channels it provides depends on the configured bit-rate, whereas the standard sets a fix number of channels. MRF49XA supports Microchip proprietary sub-GHz wireless protocols.

Due to the RSSI and DQI indicators, the transceiver is able to carry out ED (*Energy Detect*) scans over the supported frequencies to operate on the least-noisy channel. This sensing capability is crucial for CR tasks.

Figure 4.4 shows the MRF49XA pin diagram, a full description of each pin is given here:

- 1. *SDI*. Digital Input. Serial data input interface to MRF49XA (SPI input signal).
- 2. *SCK*. Digital Input. Serial clock interface (SPI clock).
- 3. *nCS*. Digital Input. Serial interface chip select (SPI chip/device select).
- 4. *SDO*. Digital Output. Serial data output interface from MRF49XA (SPI output signal).

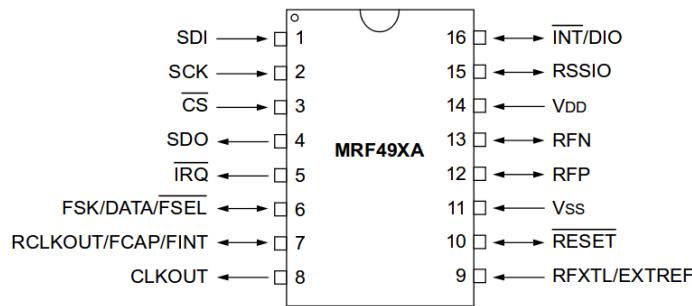


Figure 4.4: MRF49XA pin diagram, obtained from [100].

- 5. *IRO*. Digital Output. Interrupt Request Output: Receiver generates an active-low interrupt request for the microcontroller on the terminated events. Some of this events are:
  - Data-flow control registers notifications.
  - Negative pulse on interrupt input pin.
  - Wake-up timer time-out.
  - Supply voltage below the programmed value is detected.
  - Power-on Reset.
- 6. *FSEL*. Digital Input/Output. FIFO (*First-in First-out*) Select: Selects the FIFO and the first bit appears on the next clock when reading the Receiver FIFO Read Register. The FSEL pin has an internal pull-up resistor. This pin must be “high” when the TX register is enabled. In order to achieve minimum current consumption, keep this pin “high” in Sleep mode. This pin could provide other functionalities not used for our design.
- 7. *FINT*. Digital Input/Output. FIFO Interrupt: When the FIFO enable bit of General Configuration Register is enabled, this pin acts as a FIFO full interrupt, indicating that the FIFO has been filled to its programmed limit. This pin could provide other functionalities not used for our design.
- 8. *CLKOUT*. Digital Output. Clock Output: The transceiver’s clock output can be used by the host microcontroller as a clock source.
- 9. *RFXTL*. Analog Input. RF Crystal: This pin is connected to a 10 MHz series crystal or to an external oscillator reference. The crystal is used as a reference for the PLL (*Phase-locked loop*) which generates the local oscillator frequency. It is possible to “pull” the crystal to the accurate frequency by changing the load capacitor value.
- 10. *nRESET*. Digital Input/Output. Active-low hardware pin. This pin has an open-drain Reset output with internal pull-up and input buffer.
- 11. *Vss*. Ground. Ground reference.
- 12. *RFP*. RF. Input/Output Differential RF input/output (+).
- 13. *RFN*. RF. Input/Output Differential RF input/output (-).

- 14. *VDD*. Power. RF power supply. Bypass with a capacitor close to the pin.
- 15. *RSSIO*. Analog Input/Output. Received Signal Strength. Indicator Output: The analog RSSI output is used to determine the signal strength. The response and settling time depends on the external filter capacitor.
- 16. *nINT*. Digital Input/Output. Interrupt: This pin can be configured as an active-low external interrupt to the device. If a logic '0' is applied to this pin, it causes the IRO pin to toggle, signaling an interrupt to the external microcontroller. The source of interrupt can be determined by reading the first four bits of STSREG. This pin can be used to wake-up the device from Sleep. This pin could provide other functionalities not used for our design.

An internal transmit/receive switch combines the transmitter and receiver circuits into differential RFP and RFN pins. These pins are connected to the impedance matching circuitry (balun), and this is linked to the external attached antenna.

The data quality is checked or validated using the RSSI and DQI blocks built into the transceiver. Data is buffered in transmitter registers and receiver FIFOs. The Automatic Frequency Control feature allows the use of a low-accuracy and low-cost crystal. The transceiver is controlled via a 4-wire SPI, interrupt (INT and IRO), FSEL, FINT and RESET pins.

The complete design of the  $\mu$ Trans 434/868 is based on the implementation guidelines provided by Microchip at MRF49XA datasheet. The only difference between 434 MHz and 868 MHz version of the  $\mu$ Trans is the impedance matching circuitry facing the external antenna. The components that compose it are different for each frequency band, values are specified at table A.1 placed at Appendix A.

Current is bypassed through a capacitor and supplies the MRF49XA, a power indication LED (*Light-Emitting Diode*) is included in the design. A 10 MHz clock carries out the timing tasks. The RI hosts a  $50\Omega$  connector to which connect the corresponding antenna. 434 and 868 versions of the  $\mu$ Trans are designed to attach 434 MHz and 868 MHz external monopole antennas respectively.

The designed module allows access to some of the MRF49XA pins, not all of them. The ports accessible at the RI are Interrupt (nINT), Interrupt Request Output, SPI ports, FIFO select, FIFO interrupt, and reset. These ports enable full functionality of the MRF49XA and the remaining available pins are found dispensable.

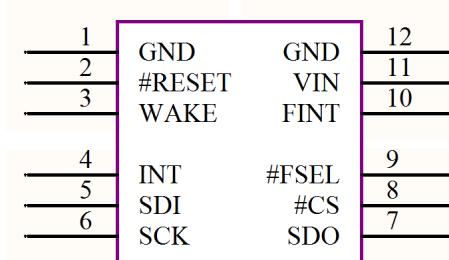


Figure 4.5:  $\mu$ Trans 434/868 component model for use.

### 4.2.2. Schematics

Schematic of  $\mu$ Trans 434/868 can be found on Appendix A at Figure A.1. Table A.1 contains the different values for the balun circuit at 434 MHz and 868 MHz.

## 4.3. Main Board - cognitiveNextGenerationDevice

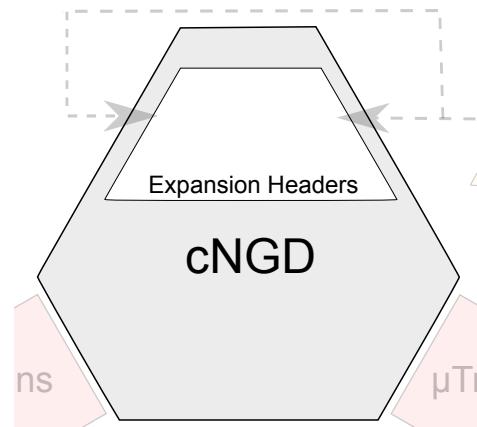


Figure 4.6: cNGD detail over global hardware modules to develop.

### 4.3.1. Description

As it was said, cNGD forms the main part of the design. It hosts main capabilities for computation, power supply, and connectivity for the platform. It has been designed seeking simplicity and functionality.

The computation unit is located at the MCU, taking control over the other modules and running the software. This MCU can be programmed through its PGE port thanks to a PGE module that interfaces it.

As part of the main goal, the firmware reviewed in section 3.3 will be used. It supposes an approach to the requirements achievement since it gives support for three interfaces, it supposes a reduction on the computation load and therefore, on the needed resources and power consumption. Moreover, it will facilitate the development, offering a stable firmware version to work over and avoiding software development tasks. This fact will force the design to keep one architecture compatible with the firmware, hence a 32-bit Microchip PIC is required as microcontroller.

Communication and control over the RIs is made through SPIs and a few more control signal later described.

Power supply will be able to take place through three ways, a  $\mu$ USB, a block terminal and a DC connector. The two first options will accept a 5 V supply, whereas the third one will take 3.3-3.6 V. This last option is thought to be supplied by a battery.  $\mu$ USB option serves as serial communication chance, and since USB provides 5 V power supply, this is employed. A second 5 V supplying connector opens possibilities to any external power supply source. A software-driven power supply system to the RIs allows the application to control the power supplied to these modules.

The device offers serial communication through the already named  $\mu$ USB options.

On the other hand, it gives access to different peripherals through a pair of 20-pin headers. The list of approachable peripherals and pins covers battery connection (for charging purposes), GPIO (*General Purpose Input-Output*)s, MCLR (*Master Clear*) pin, external interruptions, analogue inputs, USB, Ethernet module,  $I^2C$  bus, UARTs and SPIs. These options give flexibility, modularity and versatility to the device. It provides possibilities for multiple kind of applications and opens the design to new implementations and extensions for particular applications. In addition, three LEDs and two push buttons give chance to the user to control the application if required.

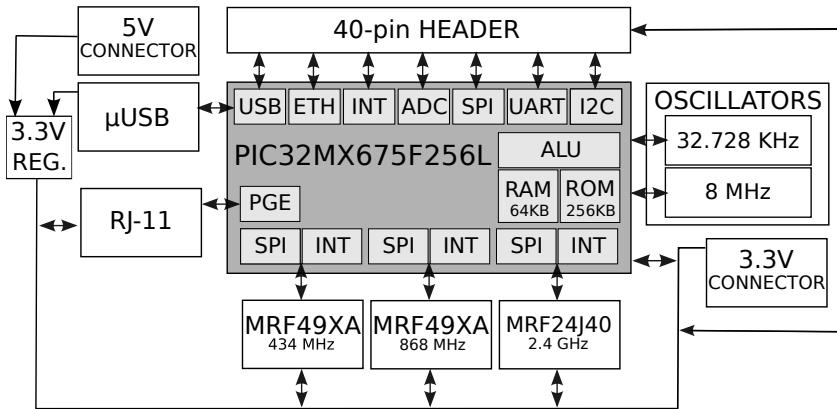


Figure 4.7: Architecture model of the cNGD.

#### 4.3.2. Schematics

Schematics can be found all together on Appendix A for better convenience. Figure A.2 gathers all the modules into one global scheme. Figure A.3 shows the MCU connections schematic. Figures A.4, A.5, and A.6 focus on 434MHz, 868MHz and 2.4GHz RIs interfaces, respectively. Figure A.7 gives the schematic for the power supply system. Expansion headers schematic is on Figure A.8, whereas Figure A.9, and A.10 illustrate the oscillator and programmer modules.

#### 4.3.3. Microcontroller

Core unit is a PIC32MX675F256L [90], a mid-range 100-pin microcontroller from the mentioned 32-bit PIC family. Being the simplest of the options that meets memory performance and peripherals minimum requirements (Table 3.1). On the other hand, its 100 pins allow a better access to the included peripherals. A larger memory version of this, PIC32MX675F512L, is pin-compatible to the firstly mentioned. This fact gives the chance to swap among these two MCUs if larger memory was needed. Main technical features follow:

- Operating Voltage Range: 2.3-3.6 V.
- 256/512 KB Flash (PIC32MX675F256L/PIC32MX675F512L). 64KB RAM.
- RUN, IDLE, and SLEEP modes. Multiple switchable clock modes for each power mode. Maximum speed is 80 MHz.

- Serial Communication Modules allow flexible UART/SPI/I2C configuration. 6 UARTs, 4 SPIs, and 5 I2Cs included.
- USB 2.0 On-The-Go peripheral with integrated PHY.
- 10/100 Ethernet MAC with MII (*Media Independent Interface*)/RMII (*Reduced Media Independent Interface*)<sup>3</sup> interfaces.
- 2 wire programming and debugging interface.
- Five 16-bit Digital Timers embedded.
- 16 channel 10-bit ADC.

Figure 4.8 shows a block diagram of the PIC32 family. Distinct 32-bit buses for peripherals and system can be observed. Moreover, DMA (*Direct Memory Access*), RAM, and flash controllers, together with external modules, take place on it.

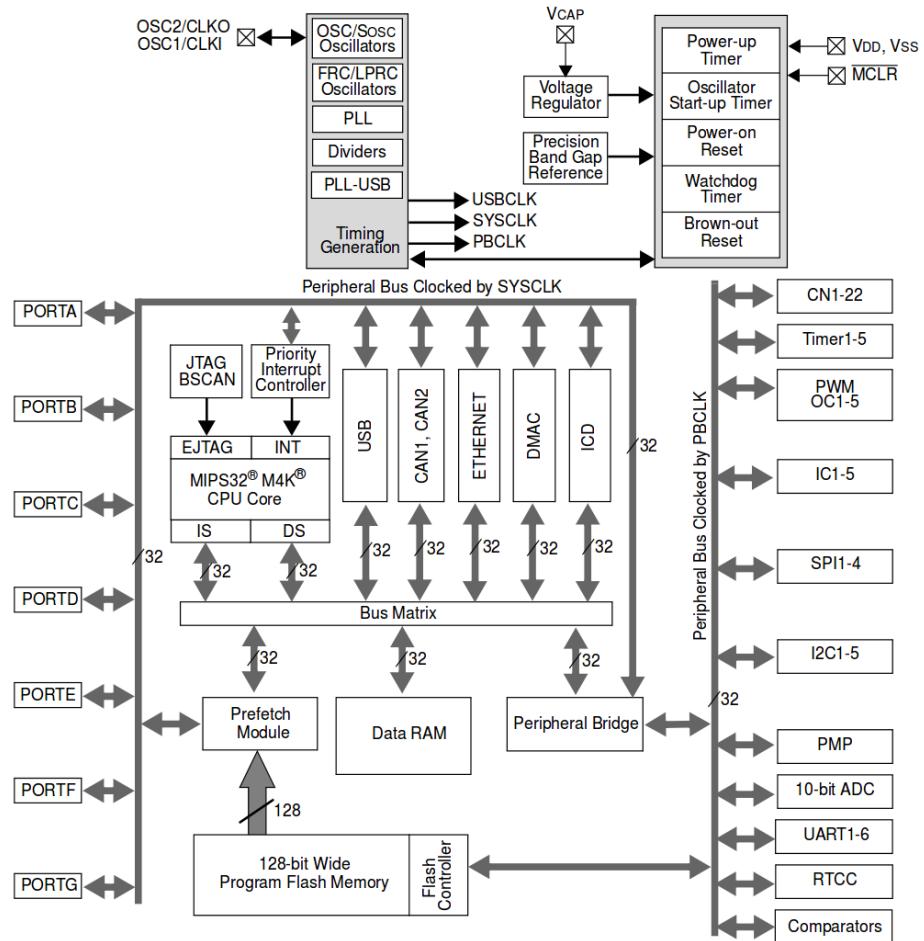


Figure 4.8: PIC32 family block diagram, obtained from [90].

The whole MCU can be divided into four blocks: core, memory, integration system, and peripherals:

<sup>3</sup>Media Independent Interface and Reduced Media Independent Interface are standard interfaces used to connect a Fast Ethernet MAC-block to a PHY chip.

- The main core is based on a RISC MIPS M4K. This module takes control over the system. It controls the peripherals and executes the programmed code.
- A further observation into the memory system allows differentiate among Flash, RAM, registers, and boot flash. The system loads the data on a cache memory and prefetches the instructions. Figure 4.9 shows the block diagram of the prefetch module.

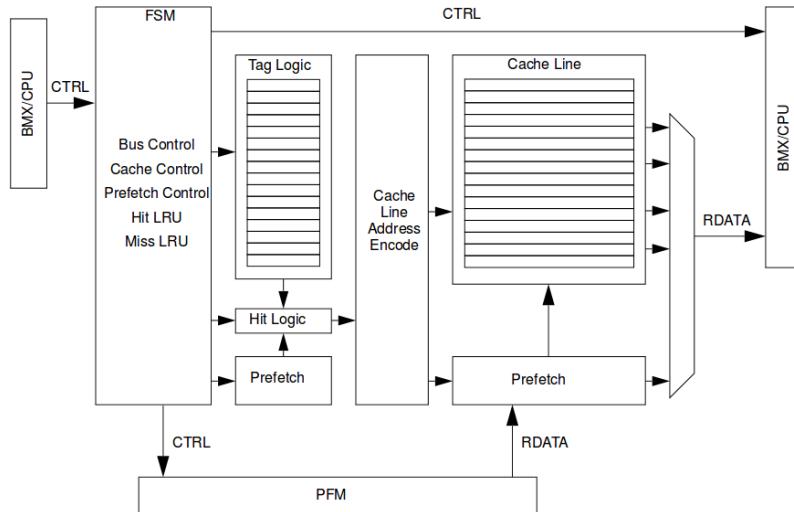


Figure 4.9: Prefetch cache module block diagram, obtained from [90].

- Integration system is compound by the set of modules needed to make work the core and peripherals. These are interrupt control, oscillator system, reset, watchdog timer, and power-up timer. For detailed explanation of each module consult the microcontroller manual [90].
- Peripherals are a set of independent modules oriented to perform specific tasks. Here, it is provided a brief description of the main peripherals and the use they receive in the system:
  - SPI.** The SPI module is a synchronous serial interface that is useful for communication with external peripherals and other microcontroller devices. These external devices may be serial EEPROM (*Electrically Erasable Programmable Read-Only Memory*), shift registers, display drivers, etc. At cNGD, three SPIs are used to set communication with the RIIs. Another SPI is linked to the expansion headers for general purposes.
  - UART.** This peripheral is a serial I/O module. The UART is a full-duplex, asynchronous communication channel that communicates with peripheral devices and personal computers through protocols, such as RS232, RS485, LIN 1.2 and IrDA (*Infrared Data Association*). Two UART modules are available at the expansion headers. RS232Shield described at section 4.4 uses one of this modules to set an RS232 serial interface.

- *USB OTG* (On-The-Go). The USB module contains analog and digital components to provide a USB 2.0 full-speed and low-speed embedded host, full-speed device or OTG implementation with a minimum of external components.

This USB peripheral is used to establish serial communication with other devices, cNGD it thought to include this serial communication way. Its ease of use, popularity, makes it a good choice for this purpose. On the other hand, it is interfaced into the expansion headers for generic future uses.

- *I<sup>2</sup>C*. The *I<sup>2</sup>C* module provides complete hardware support for both Slave and Multi-Master modes of the *I<sup>2</sup>C* serial communication standard. This module is commonly used to establish communication among sensors and microcontroller. One *I<sup>2</sup>C* bus can be accessed over the expansion headers.
- *TIMERS*. Four synchronous, one asynchronous, 16-bit timers that can operate as a free-running interval timer for various timing applications and external events counting.

These timers, properly configured, will drive in time the firmware operation. This affects, pretty much, all the tasks carried out by the node. Timers also must be available to drive in time the application if was required.

- *ADC*. It converts a continuous voltage quantity to a digital number that represents the quantity's amplitude.
- *ETHERNET*. The Ethernet controller is a bus master module that interfaces with an off-chip Physical layer to implement a complete Ethernet node in a system.

This module is accessible at the headers. It is essential to establish 802.11 communications, which is a desirable feature for WSN coordinator nodes or gateways.

- *GPIO*. General purpose I/O pins are the simplest of the peripherals. They allow the MCU to monitor and control other devices. Basically, almost every pin at the MCU can act as a GPIO if no other peripherals are active and multiplexed on such pin.

Several GPIOs can be accessed over the headers for general purposes.

- *RTCC* (Real-Time Clock/Calendar). It is intended for applications in which accurate time must be maintained for extended periods of time with minimal or no CPU intervention. Low-power optimization provides extended battery lifetime while keeping track of time.

This module is essential for accurate timing tasks, which are common in WSN for synchronous processes over the network. Synchronization helps saving energy since it allows better efficient communication and general operation.

#### 4.3.3.1. Pin-out

Here, multiplexed functionalities available on each pin at the MCU are described. Bold functionalities are functionalities available for use, not bold functionalities are not usable at the current design. In addition, the module where the pin is used and

the linked signal are also described. For further details about peripherals and pin descriptions see the PIC32 family reference manual [90].

PIN	PERIPHERALS	MODULE	USE
1	AERXARR/RG15	-	-
2	VDD	Power Supply	
3	PMD5/ <b>RE5</b>	Header	p23
4	PMD6/ <b>RE6</b>	Header	p35
5	PMD7/ <b>RE7</b>	Header	p22
6	T2CK/ <b>RC1</b>	Header	p36
7	T3CK/RC2	-	-
8	T4CK/RC3	-	-
9	T5CK/ <b>SDI1</b> /RC4	RI 434MHz	SPI SDO
10	<b>ECOL/SCK2/U6TX/U3nRTS/PMA5/CN8/RG6</b>	Header	p37
11	<b>ECRS/SDA4/SDI2/U3RX/PMA4/CN9/RG7</b>	Header	p21
12	<b>ECRSDV/SCL4/SDO2/U3TX/PMA3/CN10/RG8</b>	Header	p38
13	<b>nMCLR</b>	PGE/Pushbutton	Reset
14	<b>EREFCLK/nSS2A/U3nCTS/U6RX/PMA2/CN11/RG9</b>	Header	p20
15	VSS	Ground	
16	VDD	Power Supply	
17	TMS/RA0	-	-
18	AERXD0/ <b>INT1</b> /RE8	RI 434MHz	Interrup.
19	AERXD1/ <b>INT2</b> / <b>RE9</b>	Header	p00
20	<b>AN5/C1IN+/VBUSON/CN7/RB5</b>	Header	p01
21	<b>AN4/C1IN-/CN6/RB4</b>	Header	p39
22	<b>AN3/C2IN+/CN4/RB3</b>	Header	p02
23	AN2/C2IN-/CN3/RB2	-	-
24	<b>PGEC1/AN1/CN2/RB1</b>	PGE	PGEC
25	<b>PGED1/AN0/CN1/RB0</b>	PGE	PGED
26	PGEC2/AN6/OCFA/RB6	-	-
27	PGED2/AN7/ <b>RB7</b>	RI 434MHz	Reset
28	<b>VREF-/CVREF-/AERXD2/PMA7/RA9</b>	Header	p18
29	<b>VREF+/CVREF+/AERXD3/PMA6/RA10</b>	Header	p19
30	AVDD	Power Supply	
31	AVSS	Ground	
32	AN8/C1OUT/ <b>RB8</b>	RI 434MHz	Wake
33	AN9/C2OUT/ <b>RB9</b>	RI 434MHz	Power Switch
34	AN10/CVREFOUT/PMA13/ <b>RB10</b>	RI 434MHz	Chip Select
35	<b>AN11/ERXERR/AETXERR/PMA12/RB11</b>	Header	p03
36	VSS	Ground	
37	VDD	Power Supply	
38	TCK/RA1	-	-
39	<b>SCK4/U5TX/U2nRTS/RF13</b>	RI 2.4GHz	SPI SCK
40	nSS34/U5RX/U2nCTS/ <b>RF12</b>	RI 868MHz	FIFO Int.
41	<b>AN12/ERXD0/AECRS/PMA11/RB12</b>	Header	p04
42	<b>AN13/ERXD1/AECOL/PMA10/RB13</b>	Header	p05
43	<b>AN14/ERXD2/AETXD3/PMALH/PMA1/RB14</b>	Header	p17
44	<b>AN15/ERXD3/OCFB/PMALL/PMA0/CN12/RB15</b>	Header	p16
45	VSS	Power Supply	
46	VDD	Power Supply	
47	AETXD0/nSS3/U4RX/U1nCTS/CN20/ <b>RD14</b>	RI 868MHz	FIFO Sel.
48	AETXD1/ <b>SCK3/U4TX/U1nRTS/CN21/RD15</b>	RI 868MHz	SPI SCK
49	SDA5/ <b>SDI4/U2RX/PMA9/CN17/RF4</b>	RI 2.4GHz	SPI SDO
50	SCL5/ <b>SDO4/U2TX/PMA8/CN18/RF5</b>	RI 2.4GHz	SPI SDI
51	<b>USBID/RF3</b>	USB/Header	USB ID/p06
52	SDA3/ <b>SDI3/U1RX/RF2</b>	RI 868MHz	SPI SDO
53	SCL3/ <b>SDO3/U1TX/RF8</b>	RI 868MHz	SPI SDI
54	<b>VBUS</b>	USB/Header	VBUS/p07

Follows on next page.

PIN	PERIPHERALS	MODULE	USE
55	VUSB	Power Supply + Cap	
56	<b>D-/RG3</b>	USB/Header	D-/p13
57	<b>D+/RG2</b>	USB/Header	D+/p12
58	<b>SCL2/RA2</b>	Header	p11
59	<b>SDA2/RA3</b>	Header	p10
60	<b>TDI/RA4</b>	LEDS	LED 3
61	<b>TDO/RA5</b>	LEDS	LED 2
62	VDD	Power Supply	
63	<b>OSC1/CLK1/RC12</b>	Oscillators	Prim. Osc. In.
64	<b>OSC2/CLK0/RC15</b>	Oscillators	Prim. Osc. Out.
65	VSS	Ground	
66	AETXCLK/SCL1/ <b>INT3/RA14</b>	RI 868MHz	Interrup.
67	AETXEN/SDA1/ <b>INT4/RA15</b>	RI 2.4GHz	Interrup.
68	RTCC/ <b>EMDIO/AEMDIO/IC1/RD8</b>	Header	p09
69	nSS1/ <b>IC2/RD9</b>	Header	p32
70	<b>SCK1/IC3/PMCS2/PMA15/RD10</b>	RI 434MHz	SPI SCK
71	<b>EMDC/AEMDC/IC4/PMCS1/PMA14/RD11</b>	Header	p27
72	<b>SDO1/OC1/INT0/RD0</b>	RI 434MHz	SPI SDI
73	<b>SOSCI/CN1/RC13</b>	Oscillators	Sec. Osc. In.
74	<b>SOSCO/T1CK/CN0/RC14</b>	Oscillators	Sec. Osc. Out.
75	VSS	Ground	
76	<b>OC2/RD1</b>	RI 868MHz	Reset
77	<b>OC3/RD2</b>	RI 868MHz	Wake
78	<b>OC4/RD3</b>	RI 868MHz	Power Switch
79	ETXD2/IC5/PMD12/RD12	-	-
80	ETXD3/PMD13/CN19/RD13	-	-
81	OC5/PMWR/CN13/ <b>RD4</b>	RI 868MHz	Chip Select
82	<b>PMRD/CN14/RD5</b>	Push buttons	S2
83	<b>ETXEN/PMD14/CN15/RD6</b>	Header	p33
84	ETXCLK/PMD15/CN16/ <b>RD7</b>	Push buttons	S3
85	VCAP/VDDCORE	Capacitor	
86	VDD	Power Supply	
87	<b>ETXD1/PMD11/RF0</b>	Header	p26
88	<b>ETXD0/PMD10/RF1</b>	Header	p25
89	ETXERR/PMD9/RG1	-	-
90	PMD8/RG0	-	-
91	<b>TRCLK/RA6</b>	LEDS	LED1
92	TRD3/RA7	-	-
93	PMD0/ <b>RE0</b>	RI 2.4GHz	Reset
94	PMD1/ <b>RE1</b>	RI 2.4GHz	Wake
95	TRD2/ <b>RG14</b>	RI 434MHz	FIFO Sel.
96	TRD1/RG12	-	-
97	TRD0/ <b>RG13</b>	RI 434MHz	FIFO Int.
98	PMD2/ <b>RE2</b>	RI 2.4GHz	Power Switch
99	PMD3/ <b>RE3</b>	RI 2.4GHz	Chip Select
100	PMD4/RE4	-	-

Table 4.1: PIC32675F256L pins use description.

#### 4.3.4. Radio Interfaces

As it was mentioned, cNGD will be able to include up to three RIs, with three consequent frequency bands to access. This fact gives a flexibility over radio communications never offered by other WSN device. Frequencies to operate over are 434 MHz, 868 MHz, and 2.4 GHz, meeting desired features.

For communication on 434 MHz and 868 MHz, ad-hoc designed RIs described in

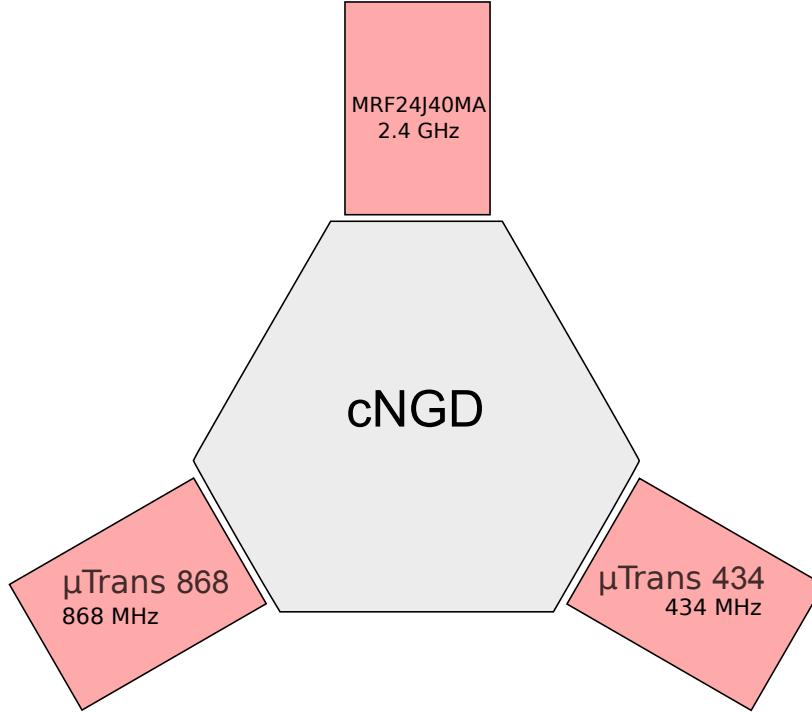
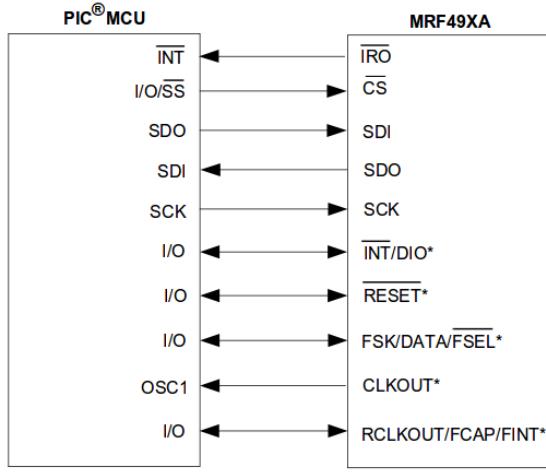


Figure 4.10: Global RI diagram.

Section 4.2 are used. These RIIs are based on the MRFX49XA radio transceiver and a full description of them can be found at such section. These RIIs share common features since the radio transceiver module is identical for both. This brings simplicity to the system. Interface between these RIIs and MCU, illustrated at Figure 4.11, are equal.



\* Implies optional signals.

Figure 4.11: Microcontroller to MRF49XA interface, obtained from [100].

Figure A.6 shows the scheme for the RI at 2.4 GHz. This is based on a MRF24J40MA [92]. This RI was already used at the FCD, and it is supported by the firmware. It is a MiWi<sup>TM</sup> transceiver also compatible with the IEEE 802.15.4 standard, this homogeneity with the other two RIIs supposes computation, cost, and power savings. It has

access over the 15 5MHz-channels the protocol enables, and employs a QPSK (*Quadrature Phase-Shift Keying*) modulation. Figure 4.12 shows the MRF24J40MA block diagram. Further details about the MRF24J40MA RI follow:

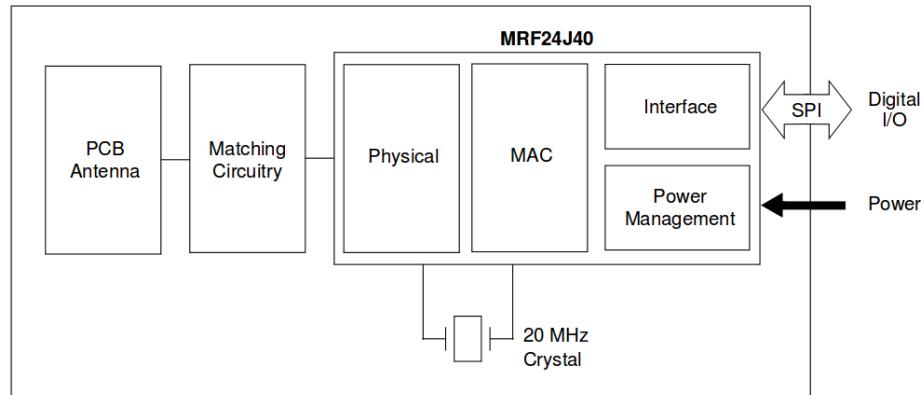


Figure 4.12: MRF24J40MA block diagram, obtained from [92].

- Operating Voltage: 2.4-3.6 V (3.3 V typical).
- Up to 120 meters range.
- Differential RF Input/Output:
  - -94 dBm Typical sensitivity with +5 dBm maximum input level.
  - +0 dBm Typical output power.
- Maximum supported data rates:
  - 250 Kbps.
  - Turbo mode enables up to 625 Kbps (not following the standard).
- Current consumption:
  - TX. I = 23 mA (typical).
  - RX. 19 mA (typical).
  - Sleep. 2 $\mu$ A (typical).
- Integrated crystal, internal voltage regulator, matching circuitry, PCB antenna, and RSSI detector.

Interface between MRF24J40MA RI is driven by 7 signals, figure 4.13 illustrates it. Serial communication is carried through a SPI, containing 4 signals. Two interruption signals for both senses and a reset signal complete the whole set.

All the described RIs are based on transceivers supporting either MiWi<sup>TM</sup> Mesh or MiWi<sup>TM</sup> P2P protocols. Figure 4.14 shows the protocol environment configuration.

A network that uses the MiWi<sup>TM</sup> protocol [47] is capable of having a maximum of 1024 nodes. Each coordinator is only capable of having 127 children, with a maximum of 8 coordinators in a network. Packets can travel a maximum of 4 hops in the network and 2 hops maximum from the PAN coordinator.

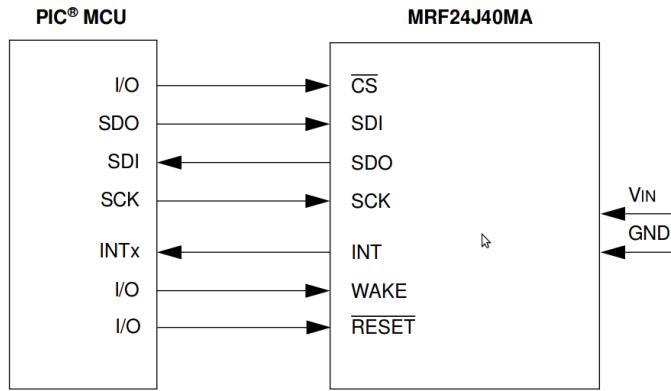


Figure 4.13: MRF24J40MA interface scheme, obtained from [92].

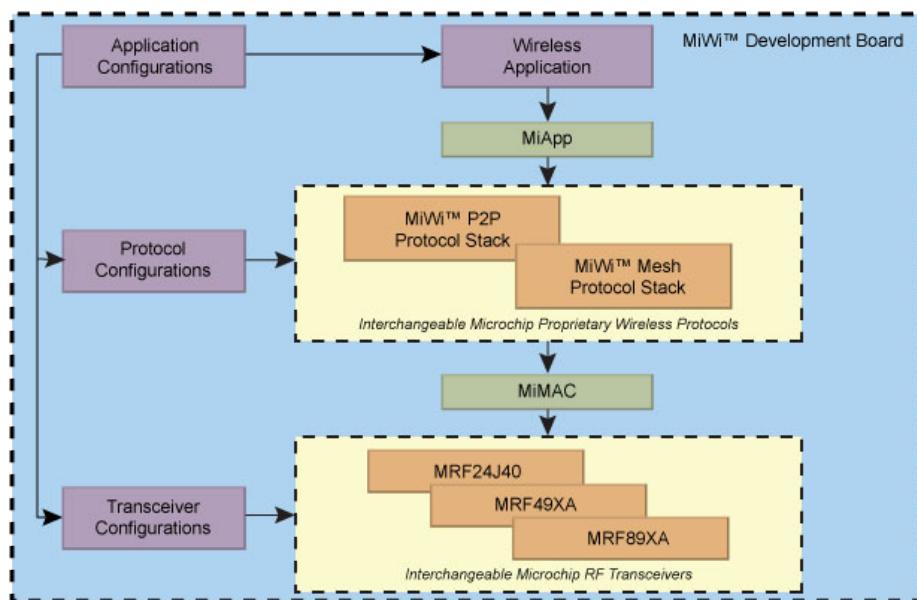


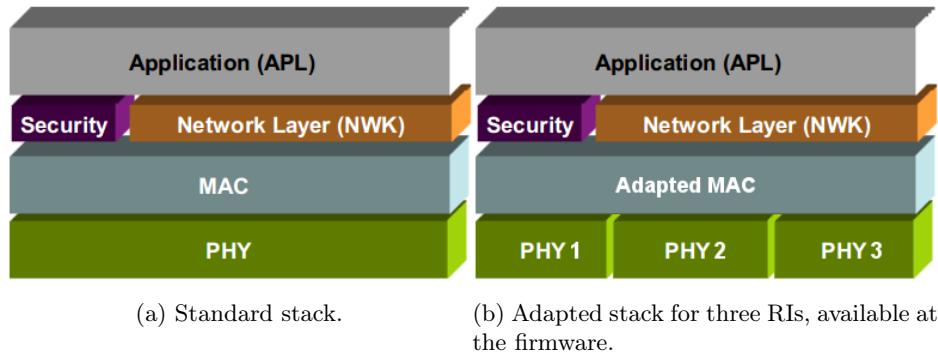
Figure 4.14: MiWi<sup>TM</sup> protocol configuration scheme, obtained from [101].

The MiWi<sup>TM</sup> P2P protocol [93] stack supports star and peer-to-peer wireless-network topologies, useful for simple, short-range, wireless node-to-node communication. Additionally, the stack provides sleeping-node, active-scan and energy-detect features while supporting the low-power requirements of battery-operated devices.

As it was mentioned, the implemented firmware offers a modification of the original MiWi<sup>TM</sup> stack that is illustrated in Figure 4.15. This adaptation of the stack is able to deal with up to three RIs.

#### 4.3.4.1. Software Controlled Power Supply

The designed system allows enabling/disabling the power supply to the RIs. This way, any RI can be completely switched off by the running application. Cutting down all possible power consumption that unused RIs could have by, for instance, running clocks.

Figure 4.15: MiWi<sup>TM</sup> Protocol.

The power control to the RIs is based on the ADG701 [102]. This is a CMOS (*Complementary Metal Oxide Semiconductor*) SPST (*Single Pole, Single Throw*) switch. This switch is designed to provide low power dissipation, low on-resistance and low leakage currents. It can operate from a single 1.8 to 5.5 V supply, which makes it ideal for use in battery-powered instruments.

Some of the features of this component are:

- 2 ohm (typical) on-resistance.
- Fast switching times: tON = 18 ns, tOFF 12 ns.
- Typical power consumption <0.01  $\mu$ W.

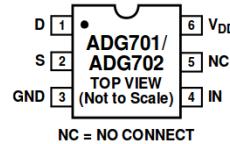


Figure 4.16: ADG701 pin diagram, obtained from [102].

Pin 4 (IN) is a digital input where the control logic signal is placed. A logic '1' will make pins 1 (D - drain) and 2 (S - source) shorted, a logic '0' will open the switch. Pins 1 and 2 can be either inputs or outputs. Pin 5 (NC) must not be connected.

In case ADG701 switch was not included into the design for any reason, optional 0  $\Omega$  resistors must be used to enable power supply at the RIs. These optional resistors are R16, R17, and R18 for 434 MHz, 868 MHz, and 2.4 GHz RIs respectively.

#### 4.3.5. Power Supply System

Power supply module is in charge of providing proper voltage and current to the cNGD. Chosen output voltage is 3.3 V, being this a normal value for CMOS family chips. The power supply system must be able to support either wired-fixed or battery options. Figure A.7, at Appendix A, shows a schematic of this module.

For external wired options, the input voltage is 5 V. This value is widely used to supply digital circuits and it is easy to find 5 V sources. Furthermore, it couples with

the USB voltage rate, being possible to supply the cNGD through a  $\mu$ USB terminal that this module includes. Apart from this USB terminal, a standard screw terminal block accepts 5 V to supply the device. This 5 V are lowered to 3.3 V by a synchronous buck regulator. This buck regulator can be deactivated for power saving when unused just switching a jumper position.

For battery options, the module includes a DC connector. This connector accepts from 3.3 up to 3.6 V. Higher voltages could damage the components since this voltage is directly injected into the circuitry to achieve higher efficiency. An included switch lets choosing the supply mode, either wired or battery.

A green LED indicates that power is applied into the circuit, on the other hand, a red LED indicates if power is supplied into the block terminal or the  $\mu$ USB terminal. This way, is easily noticeable the presence of voltage and current on the cNGD and their input origin.

The employed synchronous buck regulator is a MCP1603 [103] chip. This is a high-efficiency, fully-integrated 500 mA synchronous buck regulator with a 2.7 to 5.5 V input voltage range. The MCP1603 is available with either an adjustable or fixed output voltage. There are several available fixed output voltage options. Here, the chosen option is 3.3 V. When a fixed option is used, only three additional small external components are needed to form a complete solution, Figure 4.17.

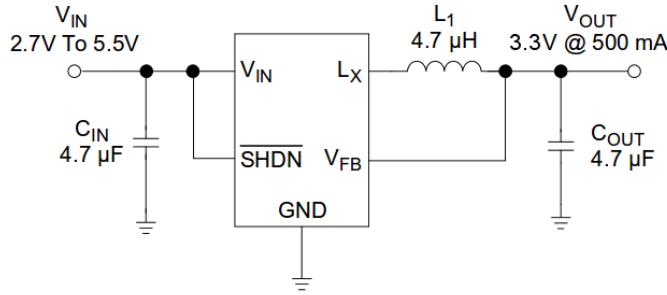


Figure 4.17: MCP1603 buck regulator configuration, obtained from [103].

It was estimated that 500 mA was enough to supply the device regarding the described maximum consumptions. Table 4.2 gives 165 mA as maximum current consumption rate. Considering a margin for current peaks at transmission and possible shields attachment, a maximum current estimation of 500 mA seems reasonable and enough for the whole platform.

Module	Max. Power
PIC32MX675F256L (Run mode)	85 mA
$\mu$ Trans 434 (TX mode)	23 mA
$\mu$ Trans 868 (TX mode)	24 mA
MRF24j40MA (TX mode)	23 mA
LEDS	10 mA
<b>TOTAL MAX CONSUMPTION</b>	<b>165mA</b>

Table 4.2: Maximum estimated power consumption of the platform.

#### 4.3.5.1. Battery

In order to give autonomy and portability to the device, a rechargeable battery system will be included. This system must keep down the global cost and size. At the same time it must offer autonomy to the device. Voltage offered by the battery system must be on the 3.3-3.6 V range.

Firstly, it is needed an estimation of platform global current consumption. This value will depends, relying strongly on the running application. However, we can assume some overestimated common values for our study. Considering two operation modes, Run and Sleep, and using values of current and operation times at Table 4.3, it results in an approximated global current consumption calculated as  $C_{estimated}$ .

Mode	Operation time percentage	Current consumption
Run	10 %	120 mA
Sleep	90 %	100 $\mu$ A

Table 4.3: Estimated values for Run and Sleep operation modes.

$$C_{estimated} = 120 \times 10^{-3} \times 0,1 + 100 \times 10^{-6} \times 0,90 = 12,1 \text{ mA}$$

Table 4.4 shows a comparison among considered battery possibilities. Other technologies and sizes, not described at the table, have been directly discarded for showing unsuitable features. An example is Ni-Cd technology batteries, which may suffer “memory effect”<sup>4</sup>, or special size batteries<sup>5</sup> whose price is extremely high. Possible options considered are some Ni-MH based and Li-ion (*Lithium-Ion*) based batteries.

Technology	Cell Nominal Voltage	Min. Num. of cells	Use complexity	Size	Capacity (mWh)	Estimated autonomy (h.)	Aprox. Unit. Cost (€)
Ni-MH	1.2/1.25 V	3	Low	AAA	600 - 1250	50 - 100	1 - 2
				AA	1200 - 2900	100 - 240	1.5 - 3
Li-ion	3.6/3.7 V	1	Medium	AAA	500 - 600	40 - 50	3.5 - 4.5
				AA	900	75	4 - 6
				CR123A	700 - 2000	60 - 165	3 - 20
				CR2	600 - 800	50 - 65	3.5
				Other	up to 3000	up to 250	40 - 60

Table 4.4: Comparison among considered batteries.

Facing similar capacities, Li-ion options show a higher cost than Ni-MH even considering that 3 Ni-MH cells are needed. In addition, Li-ion show difficulties for its use since Li-ion batteries must include circuitry to control discharge cycles and other security issues. The charging cycle is more complex as well.

The proposed solution consists of a battery consisting on 3 2900 mAh Ni-MH AA cells connected in series. This might provide up to 240 h. of autonomy considering our estimations. No more circuitry is needed for the battery except for a 3-AA battery-holder. The total cost for a complete battery pack is under 10 €.

#### 4.3.6. Timing

The microcontroller includes possibilities to properly place quartz crystals in parallel with internal amplifiers and a pair of suitable capacitors in such way they work as

<sup>4</sup>It is an effect observed in some kind of rechargeable batteries that causes them to hold less charge.

<sup>5</sup>For instance, 2/3 A, L-fat A, or CS sizes.

oscillators. The oscillator signal is then connected to the internal PLL to obtain the desired frequency. Figures 4.18 and 4.19 show the basic input scheme for primary and secondary oscillators respectively.

Primary oscillator is an 8 MHz quartz crystal used as system reference clock. The respective PLL is configured to raise this frequency up to 80 MHz. The possibility of replacing the functionality of this clock for a internal one exists, but stability and accuracy can be affected.

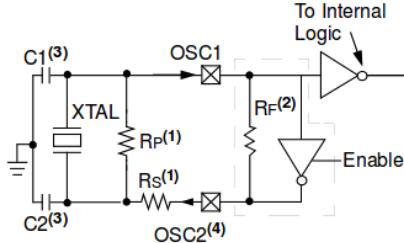


Figure 4.18: PIC32MX675F256L primary oscillator input, obtained from [90].

In order to calculate the values for  $C_1$  and  $C_2$  capacitors, we will use a common formula used for crystal-based oscillator design.

$$C_L = \frac{C_1 C_2}{C_1 + C_2} + C_P + C_I$$

Where  $C_1$  and  $C_2$  have an equal value,  $C_L$  is the load capacitance required for the crystal,  $C_P$  is the circuit parasitic capacitance, and  $C_I$  is the chip input capacitance. Typically  $C_P + C_I$  is set to 5 pF. For the 8 MHz crystal, load capacitance specified by the manufacturer is 20 pF [104]. The resulting equation remains like follows:

$$20 \text{ pF} = \frac{C_1 C_2}{C_1 + C_2} + 5 \text{ pF}$$

Solving the equation, it results that  $C_1 = C_2 = 15 \text{ pF}$ .  
Second oscillator, a 32.728 KHz crystal, supposes a low-power clock for real time clocking. It enables nodes synchronization, which is fundamental for power-saving modes.

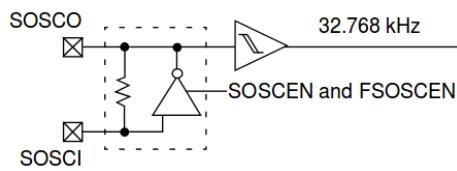


Figure 4.19: PIC32MX675F256L second oscillator input, obtained from [90].

For this second clock, load capacitance comes to be 12.5 pF [105]. Hence, the corresponding equation results as follows:

$$12.5 \text{ pF} = \frac{C_1 C_2}{C_1 + C_2} + 5 \text{ pF}$$

Resulting capacitors value are  $C_1 = C_2 = 15 \text{ pF}$

#### 4.3.7. Expansion Headers

Figure A.8, at Appendix A, shows the schematic design for this module. Basically, two headers that contain 20 pins each, enable access to unused and useful peripherals the microcontroller includes. This offers possibilities to future implementations and expansions for the node by simply attaching a device through these headers and implementing the corresponding application. Possible attachable implementations are 802.11 interfaces, data acquisition modules, external memories, extra RIIs, etc.

Here, a comprehensive description of the available peripherals and functionalities at the headers is made:

- *Power Supply*. Several pins offer power supply options:
  - VCC. Shorted to the cNGD supply system available at the platform, usually 3.3 V. Pins 14, 24, and 28.
  - GND. Shorted to cNGD ground. Pins 15, 31, and 34.
  - BAT+. Directly linked to the positive pole of the battery, thought for charging purposes. Pin 29.
  - BAT-. Directly linked to the negative pole of the battery, thought for charging purposes. Pin 30.
- *nMCLR*. Allows shields to reset the platform. Pin 08.
- *SPI*. SPI 2 is available. SDI signal at pin 21, SDO at 38 and SCK at 37.
- *UART*. UARTS 3 and 6 are available.
  - UART 3. This UART supports hardware flow control. RX signal is at pin 21, TX at 38, nRTS at 37, and nCTS is at pin 20.
  - UART 6. Does not support hardware flow control. RX is at pin 20, TX at pin 37.
- *I<sup>2</sup>C*. Two *I<sup>2</sup>C* peripherals are accessible, *I<sup>2</sup>C* number 2 and 4.
  - *I<sup>2</sup>C* 2. SCL signal is at pin 38, and SDA signal is at pin 21.
  - *I<sup>2</sup>C* 4. SCL signal is at pin 11, and SDA signal is at pin 10.
- *USB*. Fully accessible. Notice the USB can not operate over the terminal at the cNGD and the headers at once. For further information consult the manual [90]. USBID signal is placed at pin 06, VBUSON at pin 07, D+ is available at pin 12, and D- is at 13.
- *Ethernet*. It is a bus master module that interfaces with an off-chip Physical layer to implement a complete Ethernet node. Headers signals provides an RMII mode interface. Signals placement follows:
  - EMDC. (Management Clock). Pin 27.
  - EMDIO. (Management I/O). Pin 09.
  - ETXEN. (Transmit Enable). Pin 33.
  - ETXD0. (Transmit Data). Pin 25.

- ETXD1. (Transmit Data). Pin 26.
  - EREFCLK. (Reference Clock). Pin 20.
  - ECRSDV. (Carrier Sense - Receive Data Valid). Pin 38.
  - ERXD0. (Receive Data). Pin 04.
  - ERXD1. (Receive Data). Pin 05.
  - ERXERR. (Receive Error). Pin 03.
- *ADC*. 9 out of 16 available ADC channels are accessible at the headers. In addition, VREF+ is at pin 19 and VREF- is at 18, these pins have optional functionalities for the ADC peripheral.
- Channel 05. Pin 01.
  - Channel 04. Pin 39.
  - Channel 03. Pin 02.
  - Channel 11. Pin 03.
  - Channel 12. Pin 04.
  - Channel 13. Pin 05.
  - Channel 14. Pin 17.
  - Channel 15. Pin 16.
- *GPIO*. A total of 30 GPIOs are available.
- Ports A2, A3, A9, and A10 pins 11, 10, 18, and 19.
  - Ports B3, B4, B5, B11, B12, B13, B14, B15 at pins 02, 39, 01, 03, 04, 05, 17, and 16.
  - Port C1 at pin 36.
  - Ports D6, D8, D9, and D11 at pins 33, 09, 32, and 27.
  - Ports E5, E6, E7, and E9 at pins 23, 35, 22, and 00.
  - Ports F0, F1, and F3 at pins 26, 25, and 51.
  - Ports G2, G3, G6, G7, G8, and G9 at pins 12, 13, 37, 21, 38, and 20.
- *Change Notification*. A total of 9 change notification pins can be accessed. These are CN4 at pin 02, CN6 at pin 39, CN7 at pin 01, CN8 at pin 37, CN9 at pin 21, CN11 at pin 20, CN12 at pin 16, CN13 at pin 38, and CN15 at pin 33.
- *Input Capture*. Useful in applications requiring frequency and pulse measurement. Input Capture modules 1, 2 and 4 are placed at positions 09, 32 and 27.
- *Interruption*. External interruption 2 is available at pin 00.
- *Comparator*. A comparator is also available through C1IN+ at pin 01, and C1IN- at pin 39. CVREF+ and CVREF-, optional pins for the comparator, can be found at pins 18 and 19.

Notice that some of the described peripherals could be pin multiplexed and thereby, they can not be used simultaneously. For a header pin to pin description, refer to Table 4.5.

#### 4.3.7.1. Pin-out

PIN	MCU PIN	1st TYPE	2nd TYPE	3rd TYPE	4th TYPE	5TH TYPE
00	19	INT2	RE9	-	-	-
01	20	VBUSON	AN5	C1IN+	CN7	RB5
02	22	AN3	CN4	RB3	-	-
03	35	AN11	ERXERR	RB11	-	-
04	41	AN12	ERXD0	RB12	-	-
05	42	AN13	ERXD1	RB13	-	-
06	51	USBID	RF3	-	-	-
07	54	VBUS	-	-	-	-
08	13	nMCLR	-	-	-	-
09	68	EMDIO	IC1	RD8	-	-
10	59	SDA2	RA3	-	-	-
11	58	SCL2	RA2	-	-	-
12	57	D+	RG2	-	-	-
13	56	D-	RG3	-	-	-
14	-	VCC	-	-	-	-
15	-	GND	-	-	-	-
16	44	AN15	CN12	RB15	-	-
17	43	AN14	RB14	-	-	-
18	28	VREF-	CVREF-	RA9	-	-
19	29	VREF+	CVREF+	RA10	-	-
20	14	REFCLK	U3nCTS	U6RX	CN11	RG9
21	11	SDA4	SDI2	U3RX	CN9	RG7
22	5	RE7	-	-	-	-
23	3	RE5	-	-	-	-
24	-	VCC	-	-	-	-
25	88	ETXD0	RF1	-	-	-
26	87	ETXD1	RF0	-	-	-
27	71	EMDC	IC4	RD11	-	-
28	-	VCC	-	-	-	-
29	-	BAT+	-	-	-	-
30	-	BAT-	-	-	-	-
31	-	GND	-	-	-	-
32	69	IC2	RD9	-	-	-
33	83	ETXEN	CN15	RD6	-	-
34	-	GND	-	-	-	-
35	4	RE6	-	-	-	-
36	6	RC1	-	-	-	-
37	10	SCK2	U6TX	U3nRTS	CN8	RG6
38	12	ECRSDV	SCL4	SDO2	U3TX	CN10/RG8
39	21	AN4	C1IN-	CN6	RB4	-

Table 4.5: Headers pins use description.

#### 4.3.8. PGE

Figure A.10, at Appendix A, contains the schematic for the programmer module, PGE. This module allows In-Circuit Serial programming and debugging tasks over the microcontroller making use of the tools provided by Microchip. For this, PGE microcontroller ports at pins 24 and 25 are interfaced through an RJ-11 connector together with power supply pins and reset signal.

#### 4.3.9. Others

cNGD platform will include a push button to reset the MCU. It also includes three LEDs and two general purpose push buttons. They allow the user to interact with the running application if required.

- LED 1 is leaded by the signal at pin 91.
- LED 2 is leaded by the signal at pin 61.
- LED 3 is leaded by the signal at pin 60.
- Push button 1 signal is linked to pin 84.
- Push button 2 signal is linked to pin 82.

### 4.4. Serial Communication Board - rs232SHIELD

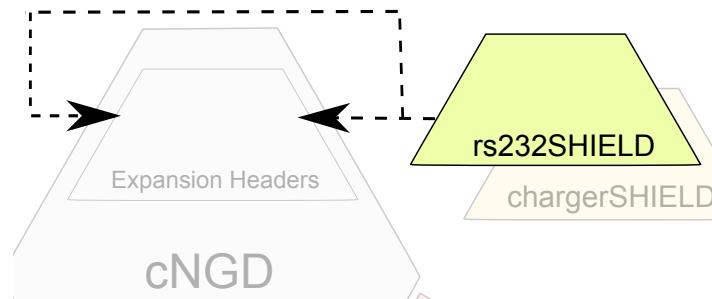


Figure 4.20: rs232SHIELD detail over global hardware modules to develop.

#### 4.4.1. Description

This extension sets serial communication between the MCU and a PC over a RS232 protocol. It supposes an alternative wired serial communication way for the included USB option, which progressively has become more popular. Including RS232 serial communication produces an increase of the cost, size, and power consumption. For this reason, it was chosen not to embed this module over the cNGD, but rather create a expansion module.

This technology could be useful for old machines, low-resources researches, or for debugging purposes during the platform development until the USB communication is fully operative. This communication way posses a simpler operation mode than USB.

UART 6 is used, accessible at the headers and not requiring hardware flow control. Since RS232 standard operates on voltage values from -12V to 12V, a level conversion is needed. This conversion takes place at the MAX3232 [106] transceiver. Figure 4.21 shows the MAX3232 scheme. TX UART signal is accessed at header pin 37, RX is at 20. Since the communication does not require for hardware flow control<sup>6</sup>, two pins are enough to set proper communication.

<sup>6</sup>Hardware flow control allows to control the serial data flow by using extra signals apart from the data signals.

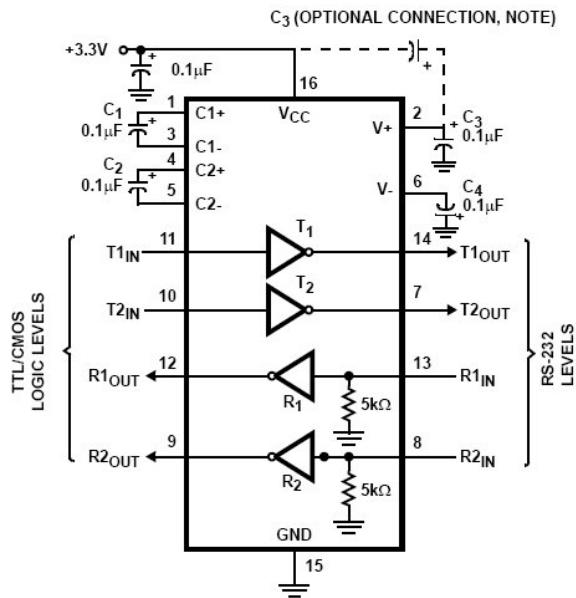


Figure 4.21: MAX3232 basic scheme, obtained from [106].

#### 4.4.2. Schematics

A schematic view of the rs232SHIELD is shown at Figure A.11 on Appendix A.

### 4.5. Batteries charger - chargerSHIELD

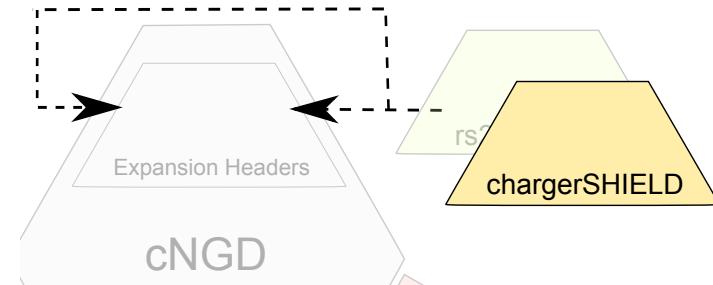


Figure 4.22: chargerSHIELD detail over global hardware modules to develop.

#### 4.5.1. Description

Ni-MH batteries are amongst the hardest batteries to charge. Whereas with lithium ion and lead acid batteries you can control overcharge by just setting a maximum charge voltage, nickel based batteries do not have a “float charge” voltage. So the charging is based on forcing current through the battery. What is usually called as linear charger.

Charge rate, usually denoted as C, signifies a charge or discharge rate equal to the capacity of a battery in one hour. For a 1 Ah battery, C is 1A. A charge rate of C/2 (0.5A), would need two hours, whereas a charge rate of 2C (2A), if supported, would

need 30 minutes to fully charge the battery. Ni-MH batteries are sensitive to damage on overcharge when the charge rate is over C/10.

Previous statements assumes that the battery has a 100% coulometric charging efficiency<sup>7</sup>. The coulometric charging efficiency of Ni-MH batteries is typically 60%. However, the faster you charge, the worse this parameter gets.

Hence, for our 2.9 Ah batteries, considering a charge rate C/10. A charge current of 290 mA could be accepted, taking from 10 to 15 hours to fully charge the batteries. However, let set the charge current around 200 mA as precautionary measure. This way, quality of the batteries is preserved and the charger is suitable for less capacity batteries.

We also want our charger to automatically switch-off once the batteries are full. We will keep sensed the voltage at the batteries poles to detect the end-of-charge moment. As initial approximation, we could establish the end-of-charge when voltage at the batteries is around 10% higher than their nominal value, 3.6 V. The sensed voltage will be compared with a reference voltage at the input of an LM311 [107] comparator. Reference voltage is obtained by a simple voltage divider.

The LM311 is a single high-speed voltage comparator. These devices are designed to operate from a wide range of power-supply voltages, including  $\pm 15$  V supplies for operational amplifiers and 5 V supplies for logic systems. The output levels are compatible with most TTL (*Transistor-Transistor Logic*) and MOS (*Metal-Oxide-Semiconductor*) circuits. These comparators are capable of driving lamps or relays and switching voltages up to 50 V at 50 mA. Figure 4.23 gives the recommended configuration provided by the manufacturer.

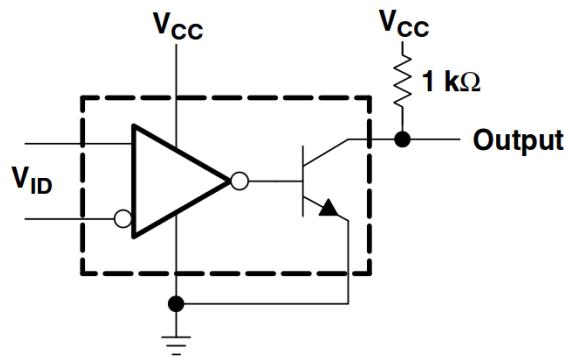
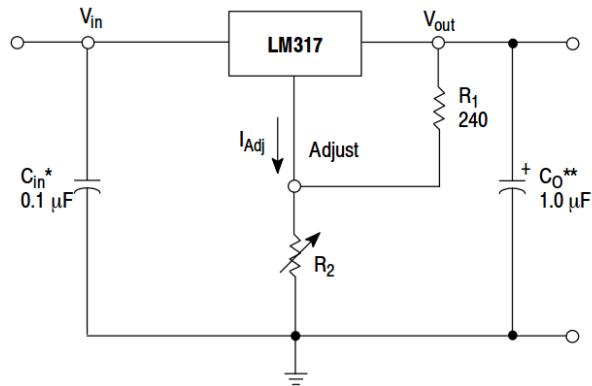


Figure 4.23: Collector output LM311 configuration, obtained from [107].

To set a fixed current through the batteries, an LM317 [108] voltage regulator is used. This is an adjustable 3-terminal positive-voltage regulator designed to supply up to 1.5 A of load current with an output voltage adjustable over a 1.2 V to 37 V range. It employs internal current limiting and thermal shutdown. Figure 4.24 illustrates the configuration required for the LM317. The formula below describes the relation among components and  $V_{out}$ .

$$V_{out} = 1.25V \left(1 + \frac{R_2}{R_1}\right) + I_{Adj} R_2$$

<sup>7</sup>Part of the electrical charge stored in a battery by charging that is recoverable during discharging. Usually expressed as a percentage.



\*  $C_{in}$  is required if regulator is located an appreciable distance from power supply filter.  
\*\*  $C_o$  is not needed for stability, however, it does improve transient response.

Figure 4.24: Configuration scheme for the LM317 provided by the manufacturer, obtained from [108].

This regulator is configured to a 1.25 V output by choosing an  $R_2$  of  $0 \Omega$ . Then a fix resistor between its output and ground will limit the output current and therefore, the input current. This fixed current will flow through the batteries during the charge. In addition, the drain of an IRF5802 [109] nMOSFET power transistor is placed at the regulator output. This means that the operation mode of the transistor affects the flowing current. If the transistor is under ohmic or saturation modes, current flows. The current does not flow if the transistor is cutoff.

IRF5802 presents a maximum drain-source on-resistance  $R_{DS(on)}$  of  $2.2 \Omega$ . Gate to source voltage  $V_{GS}$  can vary from  $-30 \text{ V}$  to  $+30 \text{ V}$ , and gate threshold voltage  $V_{GS(th)}$  is around  $4 \text{ V}$ . Maximum drain current  $I_D$  goes up to  $600 \text{ mA}$ .

The previously mentioned LM311 output controls the gate of the transistor. It switches the transistor operation modes, from ohmic to cutoff, when LM311 output is set to high or low, respectively. And the output of this operational amplifier, in turn, relies on the sensed voltage at the batteries.

A green LED driven by the comparator output, turns illuminated when this output drops enough to make current flows through the LED. This way, the LED indicates when the batteries voltage is over the reference voltage. A precautionary diode preserves the batteries from being reversely charged.

The whole system was simulated on Altium Designer and simulation traces can be observed later at this same section.

#### 4.5.2. Schematics

A schematic of charger232SHIELD can be seen at Figure A.12 on Appendix A.

#### 4.5.3. Simulation

Simulation at Altium Designer showed different behaviors facing different voltage at the batteries. It was needed to simulate models for LM311 operational amplifier and IRF5801 nMOSFET transistor. LM317AMDT was emulated composing simpler components. Figure 4.25a illustrates the output voltage at the LM311 comparator depending on the voltage at the battery. Figure 4.25b directly relates current flow over

the battery with the voltage it posses. Graphics show the right behavior, changing properly the desired voltage and current values at certain battery charge.

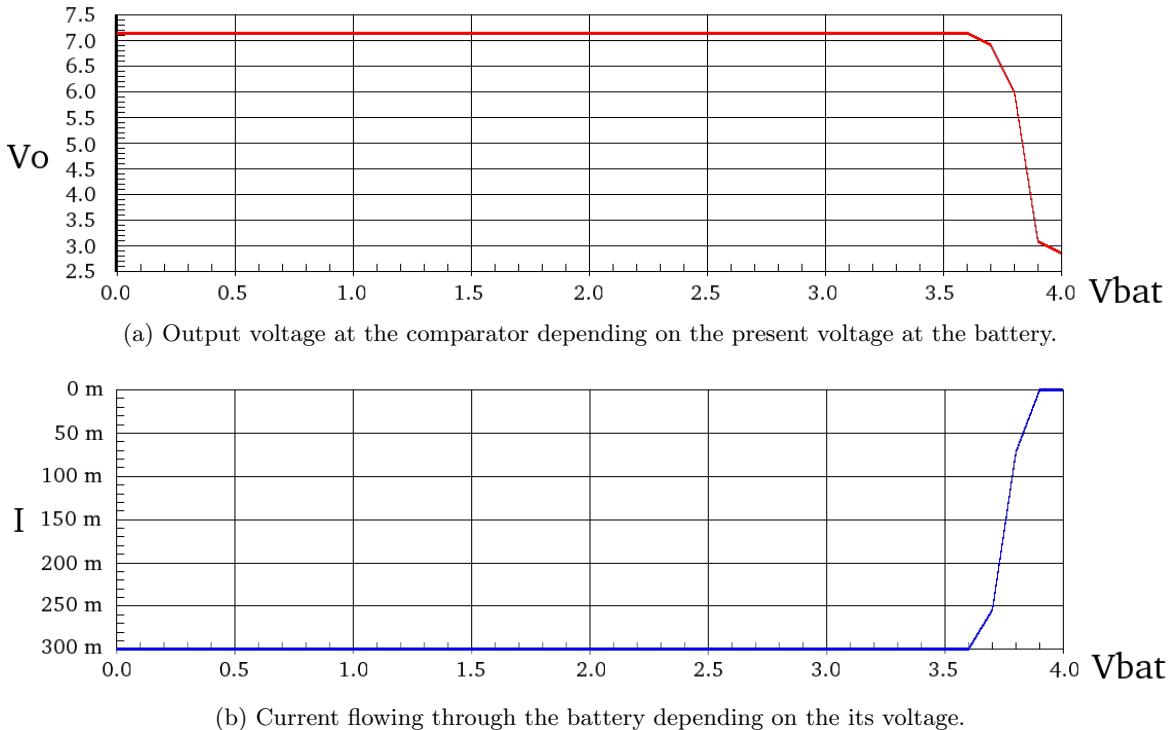


Figure 4.25: chargerSHIELD simulation results.

## 4.6. Tools

### 4.6.1. Altium Designer

For the designing task, an EDA (*Electronic Design Automation*)<sup>8</sup> software called Altium Designer will be used. Altium Designer is a software package for printed circuit board, FPGA, embedded software design, and associated library and release management automation. It is developed and marketed by Altium Limited of Australia, and is commonly used by researchers at LSI. All of the cNGD design process is carried through Altium Designer (on its Version 10).

Despite the amount of functions offered by Altium Designer, the two main options needed for this project are schematic capture, for circuit edition, and PCB design, for subsequent layout deployment.

Options provided by the schematic capture module, required for the design task, includes:

- *Component library management.* Useful to create own components facing the logic design.
- *SPICE* (Simulation Program with Integrated Circuits Emphasis) *mixed-signal circuit simulation.* In order to simulate any functionality. Specially useful in

---

<sup>8</sup>Electronic design automation is a category of software tools for designing electronic systems such as printed circuit boards and integrated circuits.

analog modules design.

- *Netlist export.* This way, it is possible to export our designs easily to process under other softwares.
- *Reporting and BoM (Bill of Materials) facilities.* For simple management purposes.
- *Multi-channel, hierarchical schematics and design re-use.* Facilitating the design task.

On the other hand, required options from the PCB design module are:

- *Component footprint library management.* Equally useful to manage new components and embed them into our PCB designs.
- *Component placement.* In order to freely design our own layouts.
- *Manual trace routing, with support for multi-trace routing.* Enabling and facilitating the layout design.
- *Interactive 3D editing of the board and MCAD export to STEP.* To obtain more realistic views and complete idea of our implementations on advance to their manufacturing.
- *Manufacturing files generation with support for Gerber formats.* Essential for manufacturing process.

## 4.7. Components Library: cngd lib

For the accomplishment of the full design of the platform over Altium Designer, a component library with all the required but not found component models was set throughout the design process.

As it was said, the library includes models from all those components needed but not found as well as modifications of existing models in order to suit the design criteria. Every contained model embeds schematic, footprint and 3D model. 3D models are useful for 3D rendering and checking proper hardware deployment. Electric models of components are not generally included.

Library is available at the mentioned repository that hosts the hardware design files. All the content included into the library is ready to be used under Altium Designer. Any use of the library is free.

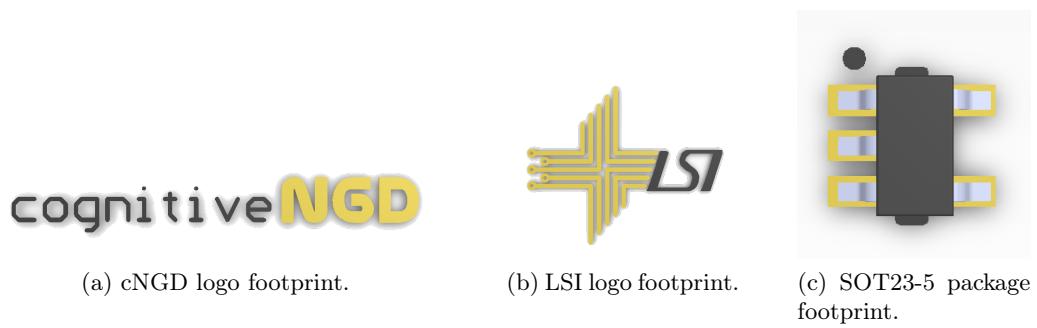


Figure 4.26: Specially designed footprints for the platform.

# Chapter 5

## Implementation Study

*Time of your life*

Thomas & Guy-Manuel, Daft Punk

In this chapter, tasks related to the physical implementation of the platform are described. These tasks cover from the layout design, 3D modeling, and GERBER files generation, to the final components mounting on the PCB. The decisions about components or layout designs, and final results are exposed.

### 5.1. Introduction

Once the system is designed over a logic circuitry level, it is moment to build a PCB design responsive to the developed schematics. This process includes the PCB shape design, components placement, tracks routing, silkscreens edition, etc. These tasks are achieved over the Altium Designer, same software described in Section 4.6.1.

All the implementations share some common features regarding their layout designs. On the one hand, the used components will keep, whenever possible<sup>1</sup>, SMD packaging. This technology gives a more reduced size and supports lower power ratings. Precisely, the SMD package used for most of two-terminal components is the 0603 size<sup>2</sup>.

All the created layouts are built over two-layer boards. Two layers are enough for our purposes since they allow to place the components in a sufficiently reduced area and power planes are not required. More layers, in addition, would increase significantly the cost of the boards.

For the tracks routing process, some considerations are taken. Ground planes are placed to face noise reduction, this way a low impedance path is provided for the power signals. Moreover, parasitic EM emissions are reduced. Power tracks are width enough to support the supplied current flow.

Together with the layout deployment, Altium Designer offers the chance to embed 3D models of each component, building up eventually a whole 3D model of the complete board. This tool was employed, and 3D models were rendered, obtaining a preview of the modules on advance to their materialization.

---

<sup>1</sup>Some components do not offer SMD (*Surface-Mount Devices*) packaging due to their electric properties.

<sup>2</sup>1.6 mm x 0.8 mm (0.063 in x 0.031 in). Typical power rating for resistors are 0.1 watt.

PCB manufacturer, PCBCART<sup>3</sup> will produce the PCBs from the outputted fabrication files. Final PCB posses a 1.6 mm thickness, and copper layers are 35  $\mu\text{m}$  thick.

Sequential mounting, and the existing firmware allow to carry through simple tests to check proper functionality and detect errors. Actually, several corrections and adjustments were made during the mounting task.

### 5.1.1. Tools

Small package components, like the employed here, requires for precision tools to be mounted. Proper tweezers are helpful dealing with tiny components. The soldering station used is a JBC AM 6000 [110], that offers several soldering tips on the 1-4 mm diameter and useful tools as hot tweezers, or desoldering accessories. Microscope is very valuable for better precision, the one used is a Bausch & Lomb StereoZoom 4 Microscope.

Components commonly include guidelines about their surrounding layout, soldering tips, or mounting processes at their datasheets. They are reviewed and taken into account for the process.

## 5.2. Transceivers - $\mu$ Trans 434/868

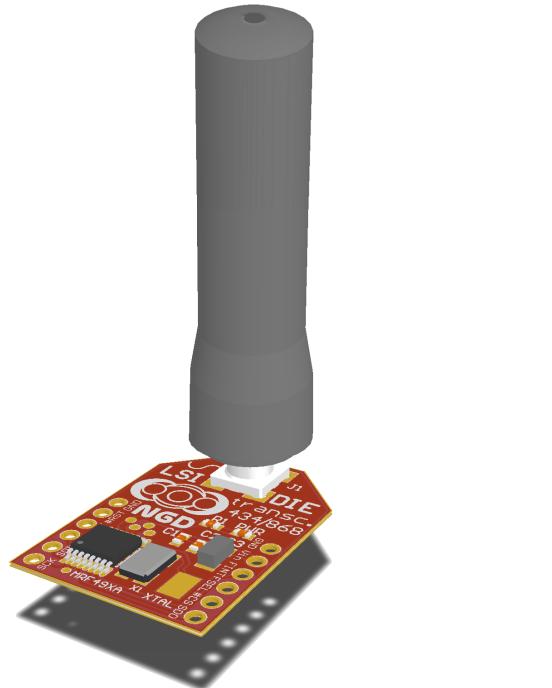


Figure 5.1:  $\mu$ Trans 434/868 3D model.

---

<sup>3</sup>[www.pcbcart.com](http://www.pcbcart.com)

### 5.2.1. Description

As it was described, this module was developed because of the non-existence of reduced size RIs including the MRF49XA transceiver. Therefore, by definition, the module must posses a reduced size. Taking advantage of the included MRF24j40MA RI, it was decided to design a PCB module fitting the footprint of this RI. This way a single footprint is common to all the included RIs. The PCB size must be suitable for this footprint, not exceeding its width.  $\mu$ Trans footprint and PCB shape are defined at Figure 5.2.

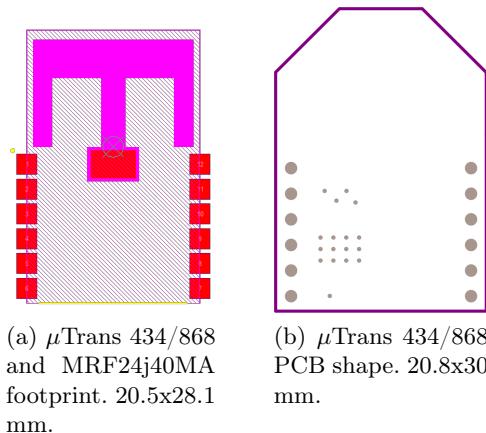


Figure 5.2: Real size footprint and shape for  $\mu$ Trans module.

Module pads are based on 1.2 mm diameter through hole vias<sup>4</sup>. These holes are width enough to let the solder tip into them and solder the pads over the footprint. These pads constitute the electrical interface between the  $\mu$ Trans and cNGD and suppose a low-cost and size solution.

MRF49XA datasheet provides hints for the inclusion of the transceiver into a PCB. It recommends to fill with ground vias the area underneath the transceiver. In addition, to have an easily accessible measure of the clock signal, the top layer posses a small pad that provides this signal. Minimal clearance among tracks and the standard track width are set to 0.2 mm. Power tracks width is set to 0.4 mm and ground planes are placed on top and bottom layers. Through holes vias minimal diameter are 0.4 mm width. Table B.1, at Appendix B, contains the needed components for the  $\mu$ Trans 434/868 on both versions.

The  $\mu$ Trans RI accepts external monopole antennas through a coaxial SMA (*Sub-Miniature version A*) female connector. Hence, antennas must posses an SMA male connector kind to be attached to the modules. Figure 5.3 illustrates the kind of connectors.

### 5.2.2. Layout

Different layers of the layout can be observed at Appendix B. Figures B.1 and B.2 show the top and bottom copper and silkscreen layers respectively.

<sup>4</sup>A via is a electrical connection between layers in a physical electronic circuit that goes through the plane of one or more adjacent layers.

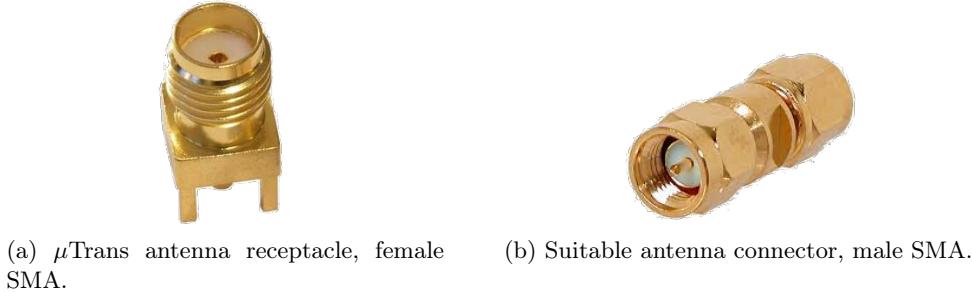


Figure 5.3:  $\mu$ Trans antenna connectors.

### 5.2.3. Prototype mounting and testing

Basic tests to check proper functionality were made while prototyping the module.

Firstly, the modules were tested over the FCD platform. For this, the  $\mu$ Trans pads were wired and connected to header prepared for the MRF49XA PICtail Plus Daughter Board. Over this platform and making use of a multi-probe oscilloscope, SPI and other interface signals were checked, and possible errors debugged.

After this, and in collaboration with other MRF49XA modules and the firmware, it was checked that the module properly carried out handshaking, network detection and joining, and data transmission and reception processes.

### 5.2.4. Final Result

Figure 5.4 shows detailed top and bottom picture of the  $\mu$ Trans prototype.

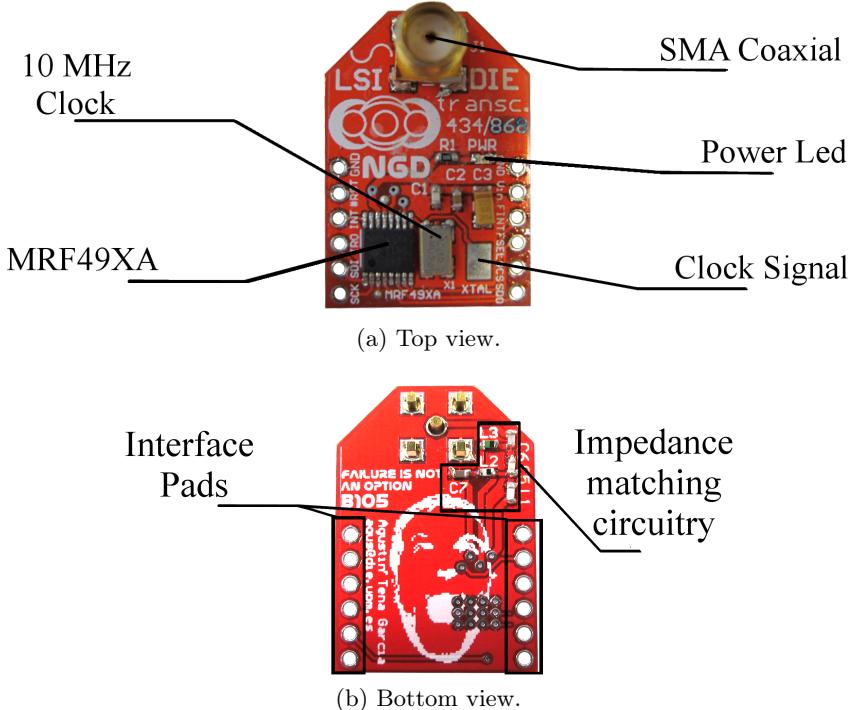


Figure 5.4: Detailed view of the  $\mu$ Trans module.

### 5.3. Main Board - cognitiveNextGenerationDevice

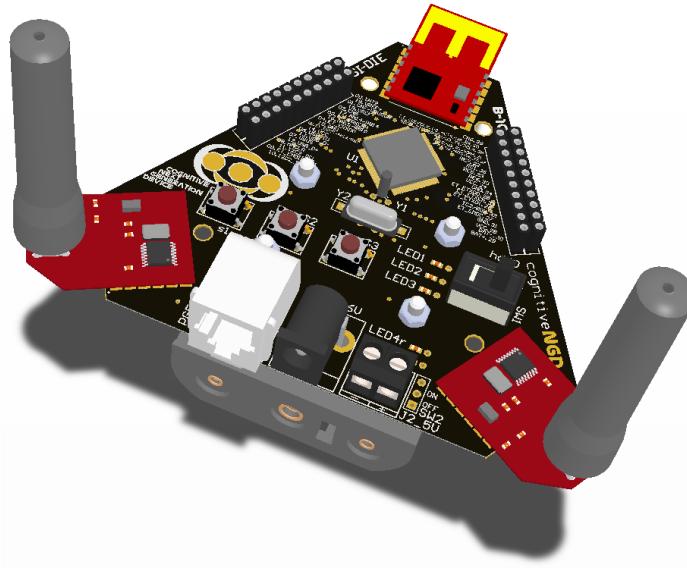


Figure 5.5: cNGD 3D model.

#### 5.3.1. Description

The cNGD PCB has to implement the whole design described in Section 4.3 attending to expand functionality and usability, and reducing size and cost. Since this is the key board of the whole system, most part of the operation requirements weight relies on it. It must mainly contain the three RI, give support to expansion modules, include the power supply and battery room, programming system, and MCU. In addition, it must provide an easy access to the control modules (LEDs and push buttons) and an easy deployment. Regarding its test-bed character, the module must include some kind of anchors to fix it anywhere or encrust some feet. All these features must be achieved keeping size and cost limited, so the module become affordable and valuable. Figure 5.6 shows the PCB shape.

Power supply system,  $\mu$ USB and PGE connector are gathered together as entire interfacing slot at the large edge 2 of the PCB. RI are thought to be placed at the three short edges. 434 MHz RI at edge 1, 868 MHZ RI at edge 2 and 2.4 GHz RI at edge 3. Figure 5.9 can give a better view of these ideas.

The board includes four holes for the battery to be attached. Figure 5.7 illustrates the way it works, four legs fix together the board and the batteries-holder.

Headers module has been placed relying on the 1 and 3 large edges, Figure 5.8a shows it. Future expansion modules will be attached to this module through exactly equal headers. This way, stacking several expansion modules ones above others will be possible. The headers footprint to include in future expansion modules, that fits the design, is illustrated in Figure 5.8b.

The minimum spacing among tracks is set to 0.1 mm around the MCU and 0.2 mm in the rest of the layout. Tracks widths vary from 0.2 mm up to 1.2 mm in power tracks. Through holes vias have a 0.4 mm width diameter. Table B.2 at Appendix B

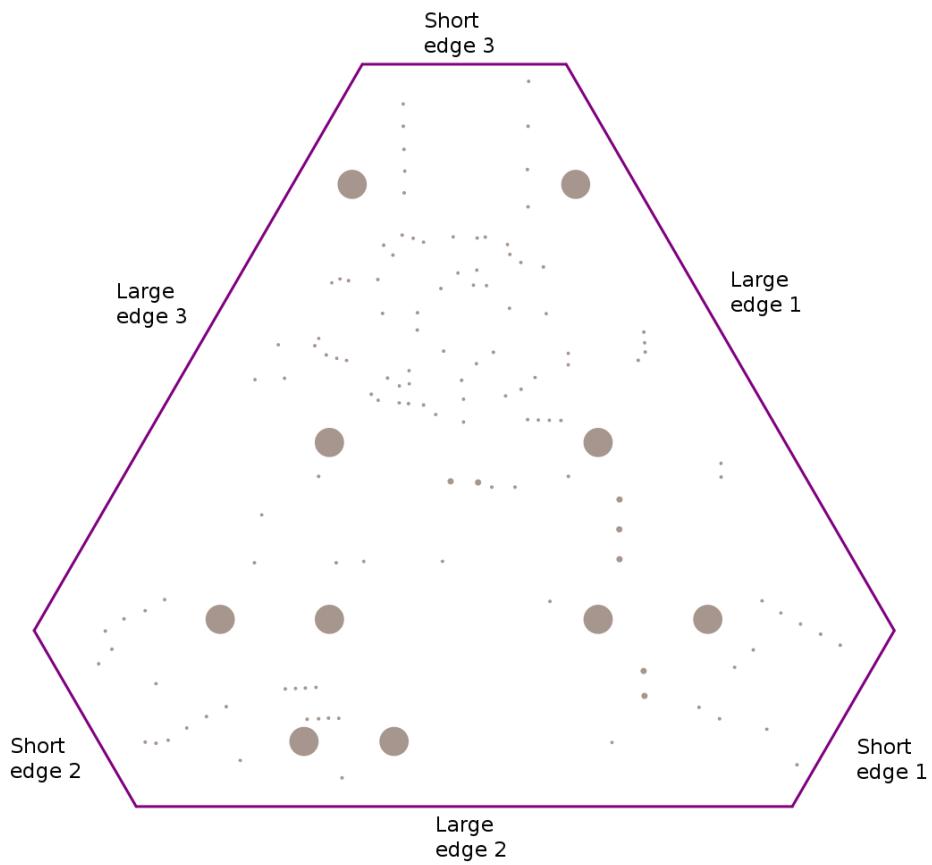


Figure 5.6: cNGD PCB real size shape. 97x84 mm

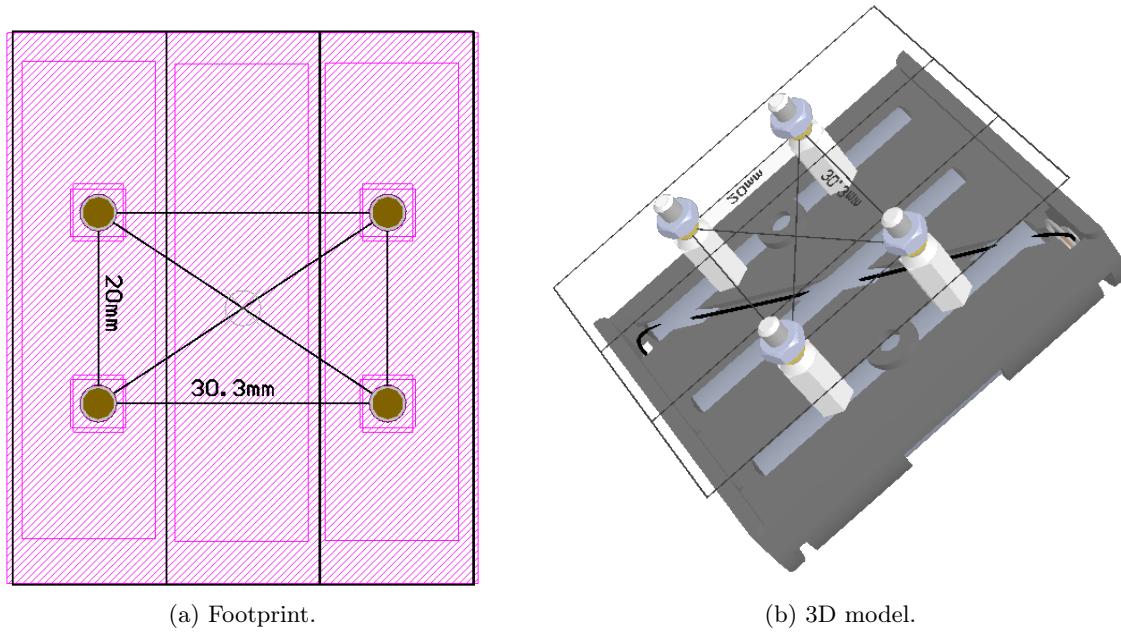


Figure 5.7: Battery anchorage system.

details the needed components for cNGD module.

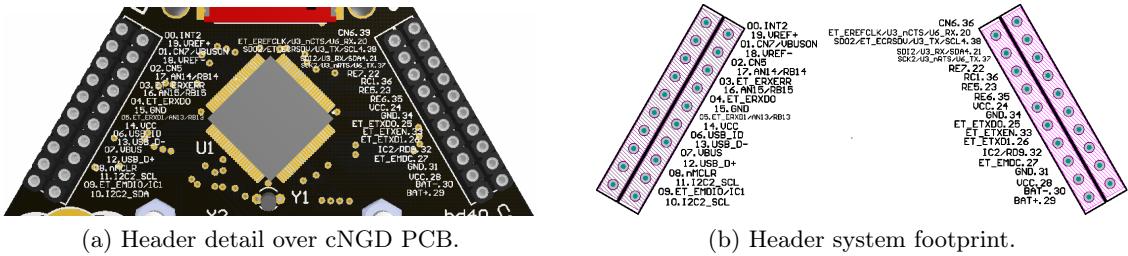


Figure 5.8: 40-pin Header system.

### 5.3.2. Layout

Different layers of the layout can be observed at Appendix B. Figures B.3 and B.4 show the top and bottom copper and silkscreens layers.

### 5.3.3. Prototype mounting and testing

When prototyping, first part to test is the right operation of the MCU. After soldering R4, C2, the reset circuit, PGE module, and the MCU itself, the microcontroller is brought under its first programming. For a right operation, is needed a right software configuration of the MCU parameters, choosing the internal clock.

Once the right programming of the MCU is tested, the external clocks are mounted and tested by changing the software parameters. After this check, it is time for push buttons and LEDs, which are tested using simple and small pieces of software. Remaining decoupling capacitors around the MCU can be soldered.

For these already described checkings, power supply from the ICD (*In-Circuit Debugger*) 3 is used. Next step is the power supply system mounting and checking by using different external sources. Since  $\mu$ USB takes part on the power supply system, it was checked at this stage. A basic test for the USB was included at the firmware.

At this point, the need of debugging traces arises to keep on mounting the device. A serial communication interface is required, and since the used firmware already supports for RS232 communication, the rs232SHIELD is mounted and employed. To read more details about this mounting process, go to Section 5.4. Headers must be soldered on the board to use this shield.

Last modules to try are the RI<sub>s</sub>. One by one, they are added to the cNGD and checked. The chosen firmware offers several radio tests for general TX/RX, handshaking, and channel switching. The RI<sub>s</sub> can be tested in conjunction with the equivalent RI<sub>s</sub> at the FCD. At this point, it is possible to check the proper operation for the software controlled switches that enable power to the RI<sub>s</sub>. Basic functions, described at Section 7.2.3 have been specifically developed to use these modules.

### 5.3.4. Final result

Figure 5.9 shows detailed top and bottom pictures of the cNGD prototype.

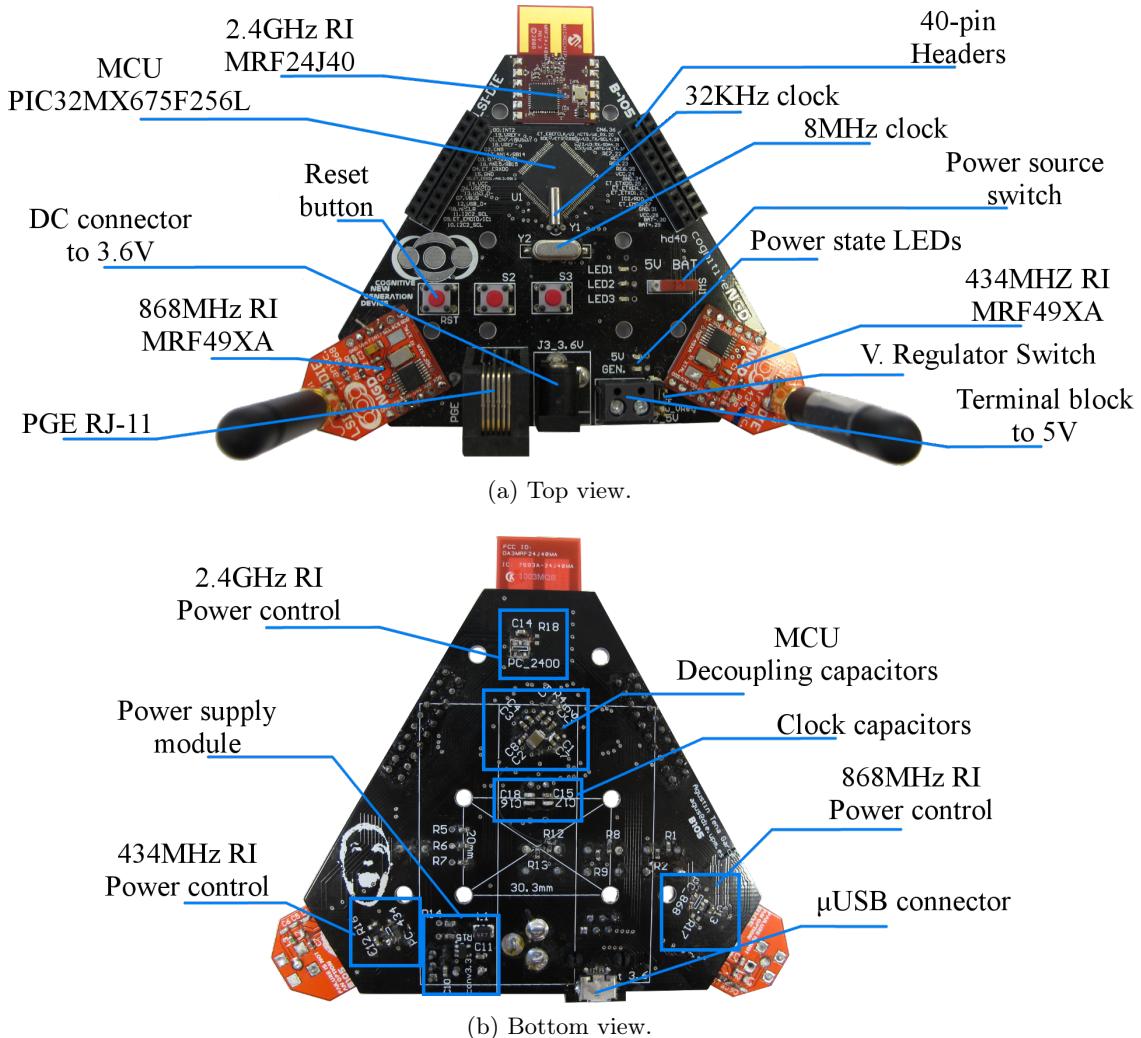


Figure 5.9: Detailed view of the cNGD module.

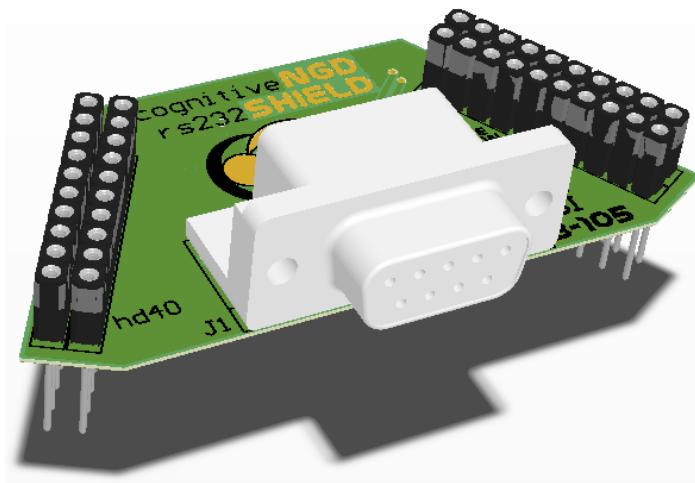


Figure 5.10: rs232SHIELD 3D model.

## 5.4. Serial Communication Board - rs232SHIELD

### 5.4.1. Description

This module is an expansion board or shield for the cNGD module. The cNGD sets the interface with the shield over the already described headers. The board shape

posses a total size of 67.9x33 mm.

The RX signal is shorted to pin 21 and TX signal to pin 38. MAX3232 chip within a SOIC (*Small Outline Integrated Circuit*) package shows suitable power and size features for this device.

In addition, the module contains a RS232 female connector to be connected to the computer or other device. Minimal track spacing is set to 0.2 mm while the tracks width is from 0.2 mm to 1.2 mm on power tracks. Vias through holes keep set to a 0.4 mm diameter. Table B.3 at Appendix B details the components needed for this module.

#### 5.4.2. Layout

Different layers of the layout can be observed at Appendix B. Figures B.5 and B.6 show the top and bottom copper and silkscreen layers.

#### 5.4.3. Prototype mounting and testing

This module offers simplicity and does not require for much testing. Basically, the whole module is properly soldered and plugged into the cNGD headers. In addition, the connection headers enable new shields stacking over. Firmware must be configured to debug over the number 6 UART.

Connecting the RS232 connector to a computer and with a suitable software, such as Minicom<sup>5</sup>, and configuring properly the communication parameters, is enough to receive data from the running application.

Serial communication parameters can be changed at the firmware configuration. However, a standard configuration follows:

- Bit rate: 115200 bits/s
- Bits per framing: 8 bits
- Parity: No
- Stop bits: 1 bit

#### 5.4.4. Final results

Figure 5.11 shows detailed top and bottom pictures of the cNGD prototype.

### 5.5. Battery charger - chargerSHIELD

#### 5.5.1. Description

This module is also a shield for the cNGD module. The cNGD sets the interface with the shield over the known expansion headers. This shield shares the same shape than rs232SHIELD.

Battery positive pole is accessible at the pin header 29, negative pole is at pin 30. These two pins are linked to the circuit. Apart from this interface, the charger encrusts

---

<sup>5</sup>Minicom is a text-based modem control and terminal emulation program for Unix-like operating systems. A common use for Minicom is when setting up a remote serial console.

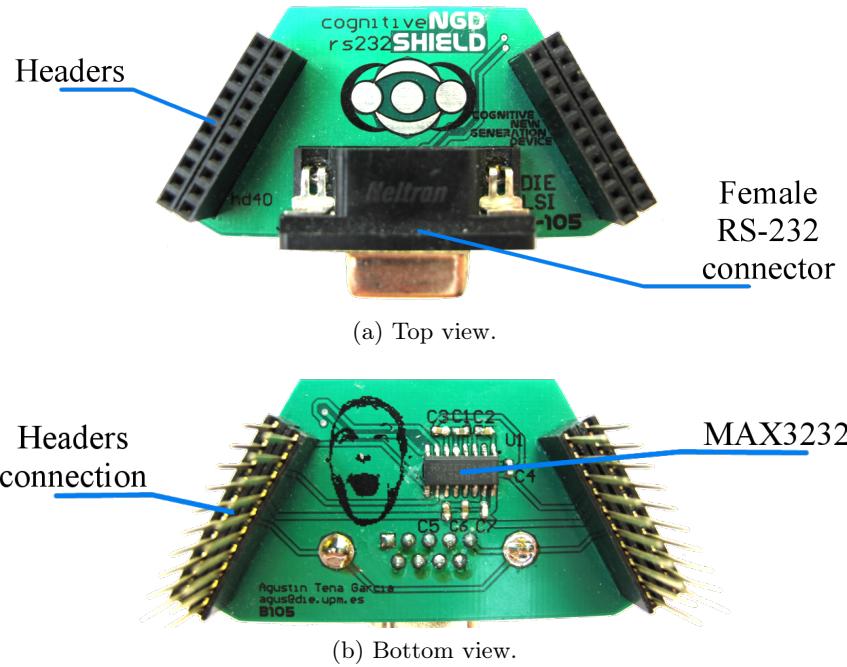


Figure 5.11: Detailed view of the rs232SHIELD module.

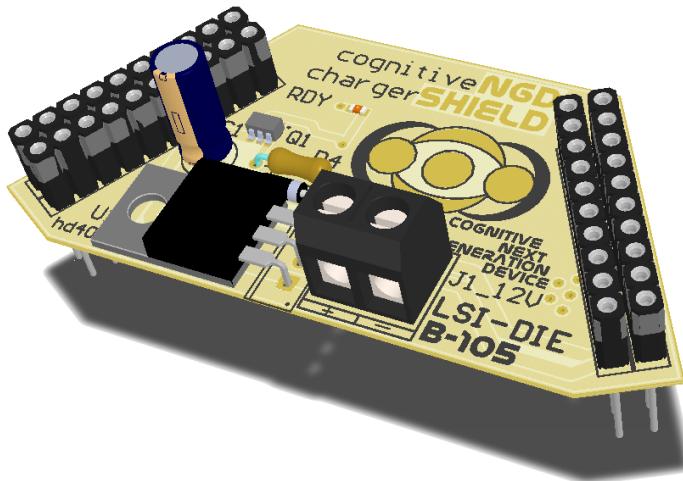


Figure 5.12: chargerSHIELD 3D model.

a terminal block to connect a external 12 V source. This source must be capable to provide up to 500 mA.

Current version of the charger does not include any heat dissipater. Tests showed that not heat dissipater was required for the current configuration. Any change over the resistors configuration must consider the overheating possibility on LM317AMDT component. Hence, a heat dissipater might be needed.

Minimal track spacing is set to 0.2 mm while the tracks width is from 0.2 mm to 1.2 mm on power tracks. Vias posses a minimum through hole of 0.4 mm diameter. Table B.4 at Appendix B details the components needed for this module.

### 5.5.2. Layout

Different layers of the layout can be observed at Appendix B. Figures B.7 and B.8 show the top and bottom copper and silkscreen layers.

### 5.5.3. Prototype mounting and testing

This module required for a lot of initial debugging, it was needed the use of a breadboard to mount in parallel parts of the module and compare behaviors. It is important to notice that some currents on the circuit are up to 250 mA, and possibilities of malfunctions that make this even higher exists. It was needed an external power supply with output current indicator and limitation for it. Of course, this module was not initially tested over the cNGD, but emulating it just with batteries or a second voltage source.

The LM311 amplifier and surrounding components, LED, R1, R2, and R3 are the first part of the chargerSHIELD to try. These parts constitute the comparator and with this, the comparator operation intends to be tested. Once this is checked, remaining of components are soldered.

At this point, a lot of tests were carried out using different voltages and resistors to calibrate the system. The goal was to adjust the current over the batteries to a proper level, measure charging times and the proper operation of the notification LED. R1, R2, R3, and R5 were finally set to their current value. Once this task was achieved, external controlled source was swapped to a commercial 12 V adapter and its proper operation was proved.

Eventually, the chargerSHIELD was tested over the cNGD.

### 5.5.4. Final results

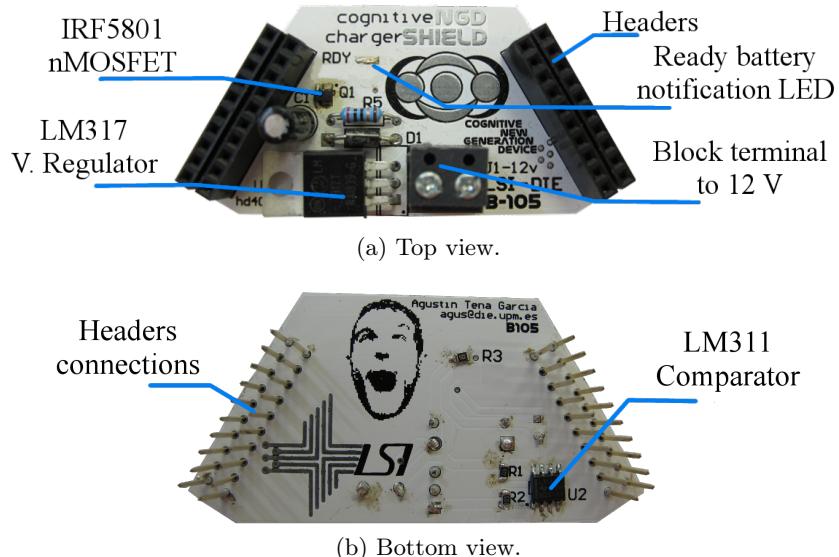


Figure 5.13: Detailed view of the chargerSHIELD module.

## 5.6. Conclusions

The whole implementation supposes a modular, versatile, functional, and flexible platform for CWSN development. cNGD module is the main piece, containing the essential functionalities for a WSN node. This main part is able to host up to three RIs, becoming the only WSN platform together with the FCD meeting this feature, and the only platform for WSNs capable to access three different frequency bands. In addition, the cNGD accepts attachable modules that brings up new functionalities without requiring a new complete design. An existing, ad-hoc developed, firmware has been used to drive the platform as a design requirement.

The development of ad-hoc RIs based on MRF49XA transceiver was needed. These developed RIs, called  $\mu$ Trans, operate over the 434 MHz and 868 MHz depending on the impedance matching circuitry and antenna they contain. They employ MiWi<sup>TM</sup> protocol and support sleep modes. The employed firmware is already prepared to fully drive these RIs. It even includes a single MiWi<sup>TM</sup> stack shared by the three possible RIs.  $\mu$ Trans are designed to be soldered into the cNGD board and they accept external antennas over a coaxial connector.

Third RI, a MRF49J40MA, consist of a commercial option that operates over 2.4 GHz. It is also MiWi<sup>TM</sup> compliant, accepts sleep modes and it is supported by the named firmware. This RI is also attached to the cNGD by soldering and it contains a PCB antenna.

cNGD is controlled by a PIC32MX675F256L. A 32-bit, low-power, high performance, 100-pin microcontroller that posses a 64KB RAM memory and 256KB flash memory. In case of need, a larger flash memory pin-compatible MCU exists. The MCU enables a clock speed up to 80MHz, including an external 8MHz clock and a low-power 32 KHz RTCC. The MCU supports sleep modes and it is widely featured.

Encrusted power supply into the cNGD has been designed to accept either 3.3 or 5 V. 5 V can be supplied over a  $\mu$ USB connector, which also serves as serial interface, or a block terminal. This voltage is lowered down to the board voltage operation, 3.3 V, by a buck regulator. 3.3 V supply option is thought for portability chances, being possible to use an included 3.6 V Ni-MH battery for the board. In addition, it is possible to control, from the MCU, the RIs power supply.

cNGD offer functionality expansions making use of a pair of 20-pin headers. These headers enable access to general use peripherals and unused pins at the MCU for future needed implementations. The board contains three LEDs and two push buttons apart from the reset push button. These components allow the user to interact with the application.

An attachable shied for the cNGD has been also developed. This enables serial communication over an RS232 protocol accessing an UART available at the header. This serial interface has an easier internal operation than the USB option and it is useful for old machines or during the platform development.

A battery charger shield was also developed in order to get the batteries easily charged through the headers. This charger automatically shuts-down the charge current at the batteries when the full charge is detected. It also contains a LED that indicates when the battery is full. Current used to charge the battery is fixed and it is set to 200 mA. The charger accepts Ni-MH batteries packs with 2000-3000 mAh capacity and varying voltages from 1 V to 5 V.

# Chapter 6

## Testings and Results

*She's up all night to the sun  
I'm up all night to get some  
she's up all night for good fun  
I'm up all night to get lucky*

Thomas & Guy-Manuel, Daft Punk

Some tests were applied to the main platform modules for the purpose of proving their proper functionality. The main capabilities that make the platform valuable such as power consumption or spectrum sensing features have been tested. The descriptions of the carried out tests and corresponding results are exposed in this chapter.

### 6.1. Test Model

The purposes of testing some of the platform functions are to prove valuable capabilities and proper modules functionality. Characterization is not expressly required and modules must respond to operation parameters provided by manufacturers. The already used firmware includes several test-benches to apply. Most of the software tools employed to test the platform are extracted, at least partially, from them. For a further description on the carried out tests, consult [9].

Tests are simple and they check bounded functions related to the platform main capabilities and features. Sleeping modes operation and corresponding current consumption at different modes was measured. Microcontroller computing capabilities and RIs performance, regarding agility or when sensing the spectrum, also have been tested. On the other hand, chargerSHIELD behavior was characterized.

### 6.2. Energy management test

This test tries to bring the cNGD under different operation modes. On the one hand, different sleeping modes and energy options for RIs and MCU are achieved consecutively. On the other hand, consumed current by the platform is measured at each situation. This test was already included at the firmware and just few changes were needed to fully carry it out. These changes suppose the functions to control the power at the RIs and they are described later in this chapter.

8 Different situations are configured. Following description covers the initial state and sequential happening events:

- A situation: Initial situation. The three RIs and the MCU are in run mode.
- B situation: 434 MHz RI goes to sleep mode.
- C situation: Power supply at 434 MHz RI is switch off.
- D situation: 868 MHz RI goes to sleep mode.
- E situation: Power supply at 868 MHz RI is switch off.
- F situation: 2.4 GHz RI goes to sleep mode.
- G situation: Power supply 2.4 GHz RI is switch off.
- H situation: MCU goes to sleep mode.

Consumption measures are reflected at Figure 6.1, where a graph shows how the current consumption goes down throughout the different presented situations.

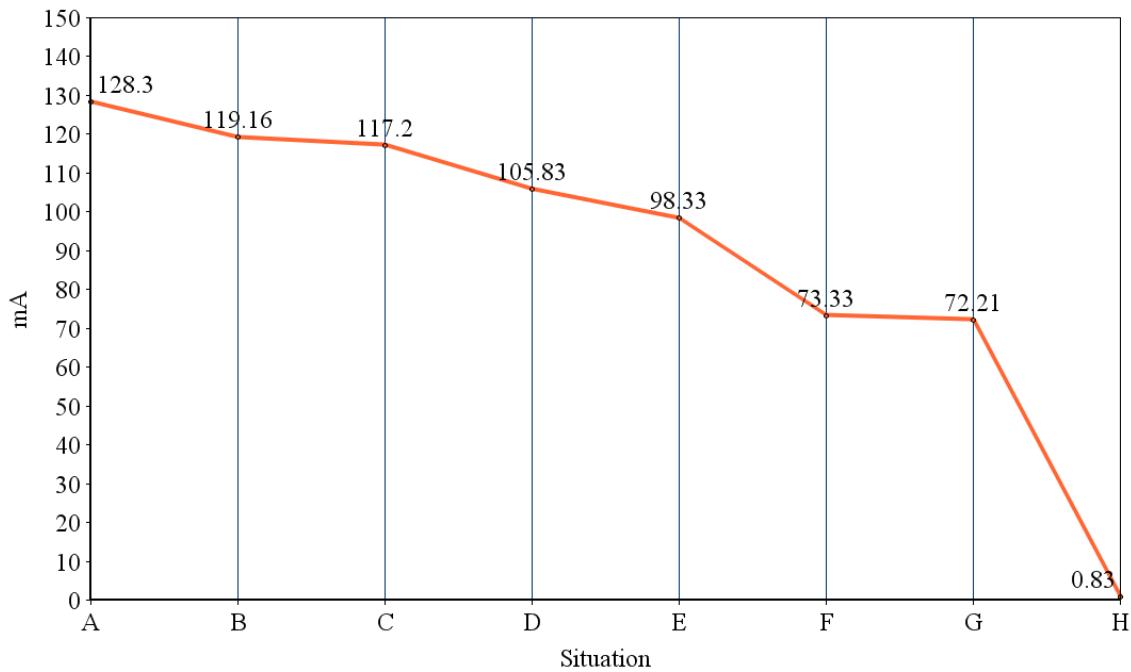


Figure 6.1: cNGD current consumption at different energy modes.

### 6.3. Computing test

This test intends to measure the computational cost of the communication protocol stack management. This could help to make a good software planning and avoid troubles interfering the application proper running. Moreover, it is important to adjust the management tasks period carefully for good communications performance.

For this, it will be determined how long takes to be accomplished the management task in a FFD (*Full Function Device*). Nominal clock frequency is set to 80 MHz and no network activity is considered. Management tasks are taken every 25 ms. This test is fully provided at the firmware.

Table 6.1 shows separately the measured time for RIs and different protocol versions, Mesh and P2P.

Management part	P2P Protocol	MiWi Protocol
Common management	134.1 $\mu$ s	150.6 $\mu$ s
434 MHz RI	1.3 $\mu$ s	1.3 $\mu$ s
868 MHz RI	1.2 $\mu$ s	1.2 $\mu$ s
2.4 GHz RI	1.2 $\mu$ s	1.2 $\mu$ s
Total time	137.8 $\mu$ s	154.3 $\mu$ s

Table 6.1: Time costs on the communication protocol management tasks.

### 6.4. Spectrum sensing test

This test tries to show the right spectrum sensing capability of the platform since it supposes a key point for its purposes. Using two cNGD prototypes and making use of the functions provided by the HAL, three simple scenarios show energy sensing features at the three different frequency bands. All the scenarios host a device continuously transmitting unicast packets at a specific channel. This RF activity is detected by the platform at different spectrum energy scans.

First scenario is conducted over the 434 MHz band. This band posses 2 available channels when using a bit-rate of 119,2 kbps. Transmitting device is making use of channel 1. Figure 6.2 shows the energy scan traces deployed by the firmware.

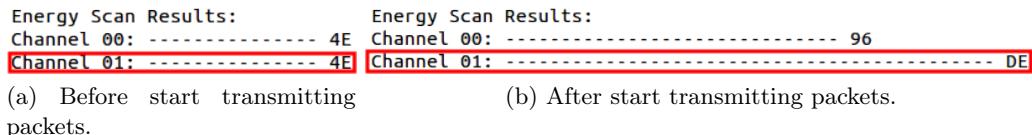


Figure 6.2: Spectrum energy sensing at 434 MHz.

Second scenario is conducted over the 868 MHz band. This band posses 7 available channels when using a bit-rate of 119,2 kbps. Transmitting device is making use of channel 4. Figure 6.3 shows the energy scan traces deployed by the firmware.

Third scenario is conducted over the 2.4 GHz band. This band posses 16 available channels. Transmitting device is making use of channel 4. Figure 6.4 shows the energy scan traces deployed by the firmware.

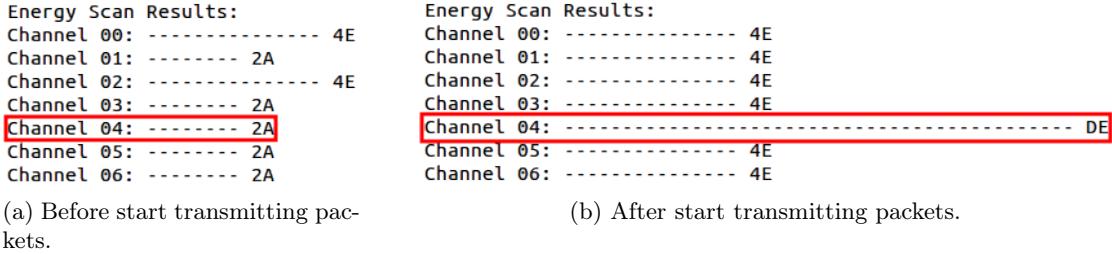


Figure 6.3: Spectrum energy sensing at 868 MHz.

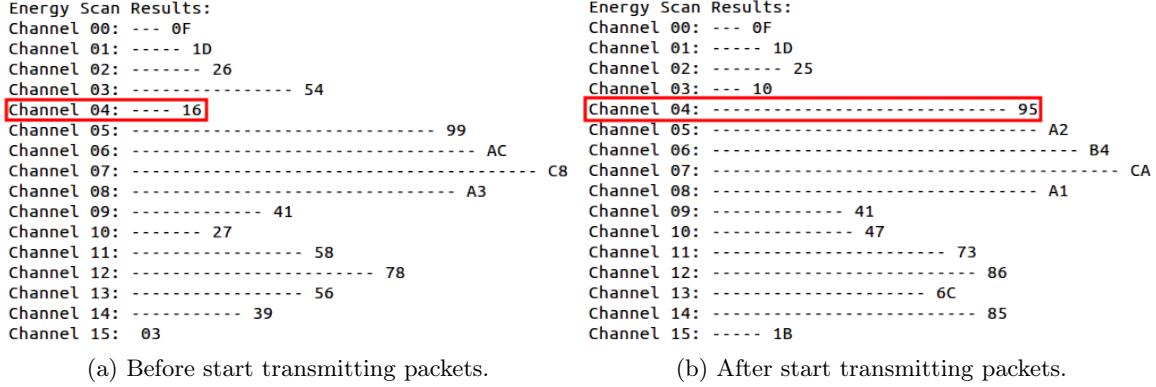


Figure 6.4: Spectrum energy sensing at 2.4 GHz.

## 6.5. Radio interfaces agility test

This test evaluates how long take different management tasks over the RIs to be accomplished. These management tasks cover spectrum energy scans and changes over TX power, operation channel, and energy modes. This test was fully available at the firmware. Table 6.2 illustrates the results.

Operation	434 MHz RI	868 MHz RI	2.4 GHz RI
TX channel switch	30.45 ms	30.4 ms	0.151 ms
TX power change	55.8 $\mu$ s	55.8 $\mu$ s	70.1 $\mu$ s
Sleep and wake up	15.4 ms	15.4 ms	0.364 ms
Spectrum energy scan	194.48 ms	908 ms	410.13 ms

Table 6.2: Time costs on the RIs management tasks.

## 6.6. Effective rate test

Effective rate provided by the MRF49XA and MRF24J40 transceivers when driven by the firmware was already described in [9]. The obtained performance must match performance achieved at the cNGD. Hence, this checking supposes a partial test compared to the full previous characterization. However, it tries new unchecked aspects of the RIs management and proves functionality.

Employed protocol at the test-bench, for simplicity, is MiWi<sup>TM</sup> P2P. The three RIs are checked and different maintenance task periods for the protocol stack are tried.

In total, periods of 1 ms, 25 ms and 500 ms were tested, not obtaining performance differences among them. To avoid redundancy, shown results correspond to 25 ms period.

The two tried TX modes are broadcast and unicast (8 Bytes address). Main communication parameters set for the test follow:

- Payload size is set to 90 Bytes.
- TX and RX buffer is configured as 10 packets for each RI.
- Total number of sent packets is 250.
- *VERIFY\_TRANSMIT* option is enabled for MRF24J40 module.
- *ENABLE\_SECURITY* option is active.

Both versions,  $\mu$ Trans module at 434 MHz and 868 MHz, posses similar performance since they share the same transceiver. Tables 6.3 and 6.4 illustrate the results at two different bit-rates, 115200 and 38400 bps, for an  $\mu$ Trans RI.

TX mode	Received packets	Total bits	TX Time (s)	Effective rate (kpbs)
Broadcast	125 (50 %)	90000	2.88	31.250
Unicast	250 (100 %)	180000	7.34	24.511

Table 6.3: Effective rate, received packets comparison for 434/868 RI using P2P protocol at 115200 bps.

TX mode	Received packets	Total bits	TX Time (s)	Effective rate (kpbs)
Broadcast	250 (100 %)	180000	6.22	28.938
Unicast	250 (100 %)	180000	8.7	20.669

Table 6.4: Effective rate, received packets comparison for 434/868 RI using P2P protocol at 38400 bps.

It is important to observe how there are not packet losses at unicast mode, and for 38400 bps there are not lost packets either when broadcasting. On the other hand, TX power and antenna conditions must be reviewed, since these suppose a change with respect to previous tests.

Table 6.5 illustrates the results for MRF24J40 transceiver as 2.4 GHz RI using a P2P protocol.

TX mode	Received packets	Total bits	TX Time (s)	Effective rate (kpbs)
Broadcast	130 (52 %)	93600	3.25	28.8
Unicast	140 (56 %)	100800	3.54	28.474

Table 6.5: Effective rate, received packets comparison for 2.4 GHZ RI using P2P protocol at 250 kbps.

A worrying issue, already described in [9], is the packet loss even transmitting at unicast mode. The reason for this fact still remains certainly unknown, but there are some possible explanations. Most feasible option at the current moment is that ACK (*Acknowledgment*) packets are misinterpreted by the receiver and it considers a single ACK response for more than one packet. This is possible because ACK packets are not numerically sequenced at MiWi<sup>TM</sup> protocol. In order to solve this issue, a work at the LSI presented in [10] shows a flow control mechanism at MiWi<sup>TM</sup> MAC layer that apparently avoid packet losses at unicast mode.

It has been concluded that the communication rate bottleneck is placed at RX rate, which sets the maximum communication bit-rate between two devices around 25-30 kbps for all the RIs.

## 6.7. chargerSHIELD performance test

As it was mentioned, small testings were made during the mounting process. Nevertheless, this test shows the chargerSHIELD performance test facing a battery full charge operation.

When testing the chargerSHIELD proper operation, few charging sequences were carried out. Taking measures every 2 hours of battery voltage conditions and the current flowing through it. Figure 6.5 shows the average results, battery voltages are view as blue bars referenced on the left side, while current flow is expressed over an orange line respect to values at the right side.

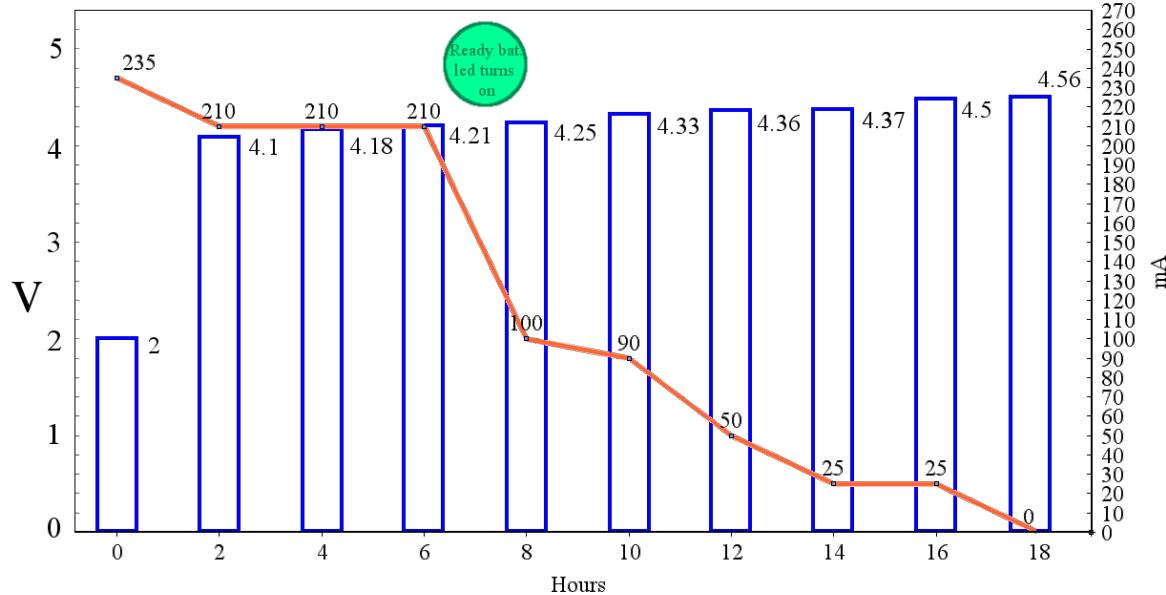


Figure 6.5: Batteries voltage evolution while charging with chargerSHIELD.

Charging time for the defined batteries can be estimated around 12-16 hours. The ready battery led notifies the battery is ready when approximately 60 % of the total charge is available, label at the figure shows the time the led turns on. It is a normal behavior for Ni-MH batteries to supply higher voltages than their nominal value despite they are supposed to provide up to 3.6 V, specially during charging processes. The

charger properly switches off the charging current when voltage at the battery reveals a high enough charge.



# Chapter 7

## Software

*At last the long wait is over  
the weight is off my shoulders  
I'm taking all control, yeah*

Thomas & Guy-Manuel, Daft Punk

This chapter gives a vision about the main changes required for the firmware to be adapted to the platform, the new software implementations and the behavior of the final demo application.

### 7.1. Tools

#### 7.1.1. MPLAB X

MPLAB X IDE (*Integrated development environment*) is a multi-platform software program to develop applications for Microchip microcontrollers and digital signal controllers. It is called an Integrated Development Environment because it provides a single integrated “environment” to develop code for embedded microcontrollers.

MPLAB X is based on the open source NetBeans IDE from Oracle. Some of its main features are:

- Supports Multiple Configurations within your projects.
- Support for multiple Debug Tools of the same type.
- Supports Live Parsing.
- Supports hyperlinks for fast navigation to declarations and includes.
- MPLAB X can Track Changes within your own system using local history.

#### 7.1.2. Programmer: ICD 3

ICD 3 is a device belonging to Microchip products that allows to get the microcontroller programmed. In addition, it enables run-time debugging using the IDE. Up to 6 breakpoints can be enabled. This feature is quite valuable regarding the platform has a development character. Further information can be consulted at the manual [111].

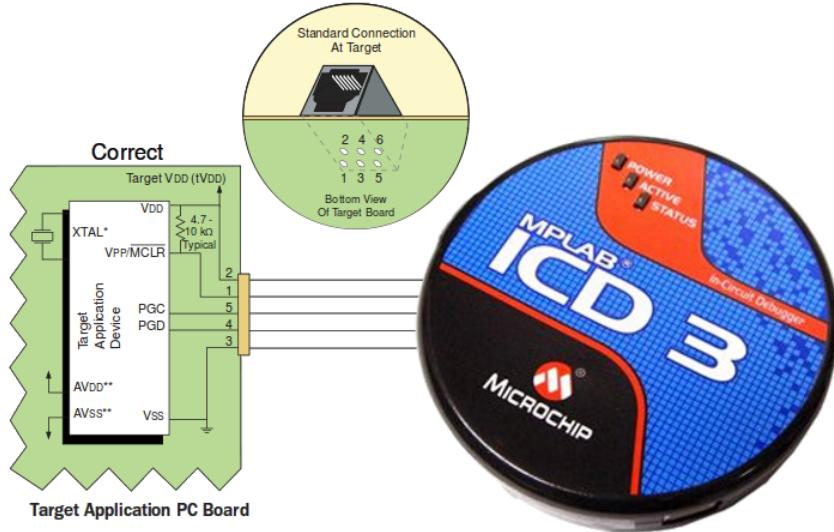


Figure 7.1: ICD3 programmer configuration.

## 7.2. Firmware and Hardware Abstraction Layer

When adapting the mentioned firmware [9] to the developed hardware platform, it was needed to deal with minor changes resulting from its novelty and also because of required hardware adaptations.

Some minor bugs that impeded the right sleeping modes operation or the spectrum sensing were fixed. As well, due to lack of tests when using three RIs at once, some software configurations were not properly set. Current firmware version corrects these issues and it is fully operative over the hardware. For the right initialization of the MRF49XA transceivers, a 50 ms delay after setting the reset signal was included.

Other adaptations taken when developing software aim at establish better ease of use. An easier configuration scheme, USB tracing modes, RI power control functions, and new global definitions are the developed functionalities.

Current and older versions of the firmware, along with small software related tests, can be found in the public GitHub repository at <https://github.com/agus-xyz/cNGD-soft>. Figure 7.2 provides a QR (*Quick Response*) code to instantly access the repository.

### 7.2.1. Platform options

When configuring the firmware to operate over different hardware platforms such as the expanded FCD version, the cNGD, or a manually configurable platform, several changes at the configuration firmware files were needed.

With the changes that the current firmware implements, adapting the firmware to any of these platforms is done by changing a single option at *Include/HardwareConfig.h* file.

- *cNGD\_PLATFORM*. Automatically sets a suitable configuration for the cNGD. *MIWI\_0434\_RI* is established over *MRF49XA\_1* transceiver, *MIWI\_0868\_RI*



Figure 7.2: cNGD software repository.

is built over *MRF49XA\_2*, and *MIWI\_2400\_RI* does it on *MRF24J40* transceiver. Remaining configurations are automatically and adequately changed.

- *FCD\_Exp\_PLATFORM*. This possibility enables the configuration to employ the FCD adapted to the lately developed expansion board. It adapts the *MIWI\_0434\_RI* over *MRF49XA\_1* transceiver, and *MIWI\_2400\_RI* at *MRF24J40*.
- *MANUAL\_PLATFORM*. This option gives freedom to the user to fully configure the employed hardware. Any of the possible Miwi<sup>TM</sup> and Wi-Fi RIs can be chosen over different transceivers such as *MRF49XA\_X*, *MRF89XA*, *MRF24J40*, or *MRF24WB0M*. Moreover, the used SPI for interfacing the RI is also configurable together with the external interruption line.

Basically, the old macros that used to be changed manually for each platform are now automatically configured when the previous options are set. Main changes take place over RI configurations and pin-outs.

Other configurable option at this same file is the possibility for debugging traces. This option is controlled enabling or disabling the *ENABLE\_CONSOLE* option. Then, within each independent platform configuration, it is possible to select the path to output traces.

### 7.2.2. USB tracing

Regarding the wide acceptance that USB protocol has achieved and how common this protocol has become, it was interesting to offer possibility to an USB console. This way, it took advantage of the included  $\mu$ USB connector. In this case, the implementation consisted on an adaptation of a Microchip USB-CDC (*Communications Device Class*) example. Stack used version is 2.9.

Adapted example offered possibilities for either polling or interruption methods, interruption management was chosen. A circular buffer would have been useful facing large volume of outputting data. Hence, at the current state, the USB tracing supposes an ideal choice for controlled environments where tracing tasks are taken carefully and

localized. Otherwise data losses or even interferences with the communication protocol management might arise.

To configure the firmware for USB traces, firstly, setting the *ENABLE\_CONSOLE* option is needed. Then, the console must be established as *DEBUG\_USB*. Configuration file for USB stack is *include/USB/usb\_config.h*

In case of need, for further information about USB stack working, the framework [112] or stack [113] manual can be consulted. For information related to USB-CDC, manual [114] might be useful.

### 7.2.3. Radio Interfaces Power Control Functions

After including power control modules for each RI at the cNGD, some software adaptations were needed to fully make them operate. For this, a pair of functions have been included at the HAL, *SwitchRION(radioInterface ri)* and *SwitchRIOff(radioInterface ri)*.

These functions set to high or low the signal that drives the power switch at each RI. Received parameter is the chosen RI. Return value might be *NO\_ERROR* or an error code.

### 7.2.4. Headers and other new definitions

Some global definitions at the firmware provide access to the pins at the headers and useful components as LEDs or buttons.

To access pins at the headers, applications can make use of the *HEADER\_XX* label definition and respective *HEADER\_XX\_TRIS* to configure the sort of pin. *XX* parameter must be replaced with the desired pin number. For instance, *HEADER\_06\_TRIS* must be set as *OUTPUT\_PIN* in order to make it a digital output. To access signal at header pin number six, the pin definition would be *HEADER\_06*.

LEDs label definitions are *LED1*, *LED2*, and *LED3* respectively. These pins are configured as output when running the node initialization.

Push buttons 1 and 2 are defined as *BUTTON\_1* and *BUTTON\_2* respectively. The firmware includes as well a pair of masks *BUTTON\_X\_PORT\_MASK* in case of these masks were required when reading whole port.

## 7.3. Demo Application Layer

Demo application layer implements a WSN application in which a transmitter and receiver module set communication and exhibit some platform communication capabilities.

Main goals for the software are to show some of the most valuable features, such as spectrum sensing and RI agility. Two devices take a role on the demo. One device will be the transmitter, configured at *include/MiApp.h* as *NODE\_1*, and the other device will be the receiver, *NODE\_2*. The application is prepared to output traces to a computer using the rs232SHIELD while running. Significant established parameters for the application are:

- P2P protocol is used for communications. The showed operation does not require routing modes.

- Bit-rate for 434 MHz RI is 19.2 kbps, and 119.2 kbps for 868 MHz RI.
- Unicast communication mode.
- The two devices posses different addresses and a common PAN (*Personal Area Network*) id.
- Sleeping modes are disabled.

The normal operation of the application responds to a sequential set of steps. Below, are shown the different behaviors and tasks throughout the execution.

- RX device is switched on. It initializes the stacks, senses the spectrum and create PANs at the most suitable channel for each RI.
- TX device is switched on. It senses the spectrum, detects active PANs and joins them.
- TX starts to send packets at 434 MHz frequency band every 2 seconds.
- At this point, antennas at 434 MHz RIs are manually removed.
- After 5 failed sending attempts, the application makes a spectrum scan over 434 MHz and 868 MHz. It detects the most suitable channel over 868 MHz and request the RX to set communication over that channel.
- When RX device asserts, TX starts sending packets at 868 MHz band.
- At this point, antennas at 868 MHz RIs are manually removed.
- After 5 failed sending attempts, the application makes a spectrum scan over 434 MHz, 868 MHz and 2.4 GHz. It detects the most suitable channel over 2.4 GHz and request the RX to set communication over that channel.
- When RX device asserts, TX starts sending packets at 2.4 GHz band.
- After 5 successful packets, the communication ends.

The demo lets show how antennas affect the sensing capabilities of the platform and the different spectrum energy scan results all over the time. It shows how the platform is able to swap communications over 3 different frequency bands using 3 different RI. All the sent packets are proved to be received when showed at the traces. The application also proves the right operation of the rs232SHIELD.

A video showing the working demo application can be accessed at <http://tiny.cc/r5p14w>. For better comfort, Figure 7.3 provides access to the video.

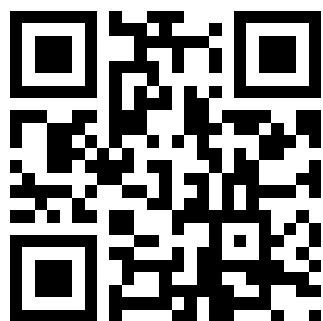


Figure 7.3: Demo application video.

# Chapter 8

## Conclusions

*Like the legend of the phoenix  
all ends with beginnings  
what keeps the planet spinning  
the force from the beginning*

Thomas & Guy-Manuel, Daft Punk

This chapter covers a full review about the whole project. A general view of the implemented system together with the main taken decisions and carried out tasks is shown. Most important conclusions are summarized and future lines are set.

The main goal of the project is to design and implement a node for the study of CWSNs. A demo application layer, also to be developed, must work integrated with an already implemented firmware. This employed firmware is able to give support to up to three transceivers sharing a single MiWi<sup>TM</sup> stack. It focus on reducing computation-costs and abstracting the developer and the application from the cognitive model and the direct hardware management.

The platform was developed responding to the requirements and a final demo lets prove the right operation of the whole system. The whole system responds to a modular design widely adaptable over the range of WSNs applications. The main module is called cNGD. This main module posses different functionalities that implements a node platform, such as power supply, power control over modules, battery, expansion headers, RIs, serial interface, and a MCU. RIs on board of cNGD offer communication over 434, 868, and 2400 MHz. RIs used for 434 and 868 MHz are ad-hoc designed RIs due to the lack of suitable size modules. This ad-hoc designed RIs are called  $\mu$ Trans 434/868 for 434/868 MHz respectively.

cNGD main module accepts expansion options over its headers, so-called shields, and for this project two shields were implemented. First shield gives chance to serial communication through an RS232 protocol. Second one is a suitable charger for the batteries. Shields allow developers to create new attachable functionalities and possibilities or give different capabilities to distinct reduced node groups. This helps to keep the downturn on complexity, consumption, size and cost but not on performance.

The sort of device shown at this project features some novel properties with respect to other CWSN devices, such as its capability to communicate over three different ISM RF bands. The hardware fits the conditions and requirements of WSN environments. Low power consumption, size and cost limitations are taken into account in order to

achieve real test-benching purposes, application development or even possible complete implementations.

In Section 1.2, a fragmented set of subgoals is described, here they follow some conclusions obtained throughout the achievement of these subgoals:

- A review all over the recent implementations related to the CWSN was made at Chapter 3. The FCD architecture is quite close to the one pursued so it is taken as guide for the designing process. Many decisions, such as a reconfiguration of the RIs and the power supply system, the design of suitable size RIs or the need for expansion options, were taken in order to face the design.
- A modular design builds up the whole system. A total of four modules were designed, a main module, an ad-hoc RI and two expansion shields that provide extra functionalities. Battery options were studied resulting in a final option of 3-cells Ni-MH battery. Components were chosen and required functionalities assigned to different modules.
- The PCB layouts were deployed and prototypes mounted, achieved size is fair for research purposes. Packaging was established regarding size and power consumption constraints. Node functionalities were checked during the mounting process making use of small pieces of software.
- The developed hardware have been submitted to several tests to check the main capabilities right performance. Hence, MCU capabilities, RI features, sleeping modes and current consumption have been proven. For this, tests included at the software have been generally used. In some cases the suffered slight adaptations.
- Firmware has been slightly modified to exploit the hardware features and the final demo application proves right operation of CR functions. Firmware adaptations and final demo application has been successfully integrated to show the platform potential.

After carrying out all the exposed tasks, conclusions about viability, value and utility about the implemented device came out. Most important points are:

- Achieved modularity with the cNGD enables robust foundations to easily build over new designs. This makes wider the possibilities for test-bed platform applications and facilitates the CWSN research.
- Flexibility is an important concept affecting both communications and applications. Three RIs make the cNGD the only WSN test-bed platform capable to access three different frequency bands. Applications might implement a great range of functionalities making use of the expansion slots that cNGD encrusts.
- Scalability is a fact since the used MiWi<sup>TM</sup> protocol provides support for variable network sizes up to 8000 nodes at its PRO version. In addition, the employed protocol gives support for both P2P and mesh topologies.
- Employed firmware supposes an efficient way to deal with multiple interfaces. A common MiWi<sup>TM</sup> protocol stack is shared among RIs and this supposes computational and memory savings. The firmware also provides a HAL to deal easily with the hardware features.

- The included MCU is a Microchip PIC32 due to firmware requirements. The model chosen supposes the less featured MCU that fulfills minimal peripherals needs and offers a wider pin-out to avoid pin-multiplexing incompatibilities.
- Power switches driven by the MCU control the power supply at the RIs. These have been shown as a really valuable function at energy saving modes. Sleeping modes are successfully implemented and allow a great autonomy using the provided battery. On the other hand, RIs have shown a very low power consumption behavior, what also helps to extend the autonomy.
- Achieved size of the platform, together with anchorage options, reveals a great usability and provide easiness facing a test-bed developments.

## 8.1. Further Studies

Regarding the future of the platform, it is important to pay attention to some points. These points have been considered as highly convenient in order to increase performance and obtaining a better utilization of the system.

- A real test-bed deployment is required to exploit the real possibilities that cNGD platform offers. Real cognitive strategies and performance must be evaluated in realistic scenarios. An implementation consisting in a standard sized WSN using cNGD nodes would be the next step to recreate a realistic CWSN. A test-bed deployment brings some necessities that are exposed throughout these future lines.
- It is important, specially when facing a test-bed deployment, to establish an OTAP (*Over The Air Programming*) system. This would facilitate the task of getting all the test-bed nodes programmed. In addition, cNGD is hardware-capable to host an OTAP system, just proper software is required.
- Since keeping an 802.11 interface is desirable in a WSN, a custom shield could give this possibility to the cNGD. Expansion options on board the cNGD gives chances for Ethernet connections. Moreover, facing the test-bed deployment, an 801.11 gateway becomes essential for some kind of IP access, control, and storage.
- CRmodule integration at the current firmware version is essential for a test-bed implementation. This module is responsible to manage all the cognition related data-flow and control. It supposes an indispensable portion to make a real CWSN.
- Some firmware modifications claims to be done. A firmware adaptation to notify a low-battery state might be an easy, valuable and useful implementation. On the other hand, a complete function to change RI bit-rates while running would be a powerful tool for CWSN investigation, and would provide extra flexibility. In addition, functions to search for PANs and establish connection to them is still needed. USB tracing modes still need for a further work, a circular buffer is needed.
- Increasing performance when sensing the spectrum would be valuable. Energy thresholds and zero-levels still need adjustments, for instance.

- Once the line of RI designing was started, there is room to keep improving the design. Giving to the current design the chance to swap its antenna impedance matching circuitry, would enable the transceiver to operate over both 434 and 868 MHz. There are not commercial tunable transceivers for WSN currently. It even might include a wake-on radio system like the one proposed in [115]. This development would allow the cNGD reduction and simplification.
- Trying new chip MRF24XA [116], launched during the development of this project, might bring a better performance on communications over 2.4 GHz. A corresponding review must be taken, specially when facing the packet losses fact at unicast mode.
- Studying different possible antennas and respective performance at  $\mu$ Trans RI might provide perspectives to increase performance and therefore, improve communications.
- Many possible implementations are possible for the expansion options at the cNGD. Typical sensor modules on WSN applications could be embedded into a shield. Different gateways or communication options such as 3G, GSM (*Global System for Mobile*) or GPRS (*General Packet Radio Service*) could be of interest for certain applications. Even developing a generic empty board with soldering slots could be a quick and cheap way to prototype shields.
- A review over the obtained performance and possibilities for less featured MCU, even changing the used architecture, might set some guidances for future improvements on the platform, despite this supposes a firmware redesign.
- Design optimizations. Current design is susceptible to suffer design optimizations. RJ-11 PGE programmer could be replaced by a simpler, smaller and cheaper option, or even could be included at the header. Creating an unified clock signal for MCU and transceivers is possible and it would save energy on clock signals generation.
- A good complement for the OTAP system could be a wireless console system. This wireless console could provide wireless tracing or even debugging to the platform. This tool would be valuable facing application developments.
- It still exists possibilities for firmware improvements. An easier access to the available peripherals at the headers might be carried out, or an expansion over the options through the USB interface.

# Chapter 9

## Estimated Costs

*Human, human, human, human,  
human, human, human, human,  
human, human, human, human,  
human, human, human after all*

Thomas & Guy-Manuel, Daft Punk

### 9.1. Physical implementation cost

Software			
Concept	Cost	Amortization	Real cost
MS Windows 7 Home	79.76	10 %	7.98 €
Altium Designer 10	195	10 %	19.5 €
Total			27.48 €

Hardware resources			
Concept	Cost	Amortization	Real cost
Personal computer	800.00 €	20 %	160.00 €
Laser printer	100.00 €	2 %	2.00 €
MPLAB ICD3 Programmer	200.00 €	2 %	4.00 €
Office consumables	50.00 €	100 %	50.00 €
Electronic working tools	50.00 €	100 %	50.00 €
Agilent E3644A Power supply	200.00 €	2 %	4.00 €
Tektronix TDS5054B Oscilloscope	10800.00 €	2 %	216.00 €
Bausch & Lomb StereoZoom 4 Binocular Microscope	324.00 €	2 %	6.48 €
JBC AM 6000 Soldering Station	2500.00 €	2 %	50.00 €
PCB manufacturing			
μTrans PCB x 20	213.31 €	100 %	213.31 €
cNGD PCB x 10	126.03 €	100 %	126.03 €
rs232SHIELD PCB x 5	95.04 €	100 %	95.04 €
chargerSHIELD PCB x 5	110.75 €	100 %	110.75 €
Electronic components	358 €	100 %	358 €
Total			1445.30 €

Labor costs			
Concept	Daily Cost	Days	Total cost
Full time engineer	113.32 €	220	24930.40 €
Social expenses (31.6 %)	35.80 €	220	7876.00 €
Total			32806.40 €

<b>Physical implementation costs</b>	
Software resources	27.48 €
Hardware resources	1445.30 €
Labor costs	32806.40 €
Printing and binding	195.00 €
Total	34474.18 €

## 9.2. Overhead and industrial profit

<b>Overhead and industrial profit</b>	
Physical implementation cost	34474.18 €
Overhead (15 %)	5171.13 €
Industrial profit (6 %)	2068.45 €
Contracted operation budget	41713.76 €

## 9.3. Fees for project management and writing

<b>Fees for project management and writing</b>	
Accumulated subtotal	41713.76 €
Writing fees (5.6 %)	2335.97 €
Management fees (5.6 %)	2335.97 €
Total	46385.7 €

## 9.4. Total budget

<b>Total Budget</b>	
Accumulated subtotal	46385.7 €
Value-added tax (21 %)	9741 €
Total	56126.7 €

The estimated cost for this project amounts to FIFTY-SIX THOUSAND ONE HUNDRED AND TWENTY-SIX POINT SEVEN euros.

Madrid, October 2013  
Project Head Engineer

Signed: Agustin Tena Garcia  
Telecommunication Engineer

# Part I

## Appendices



# Appendix A

## Schematics

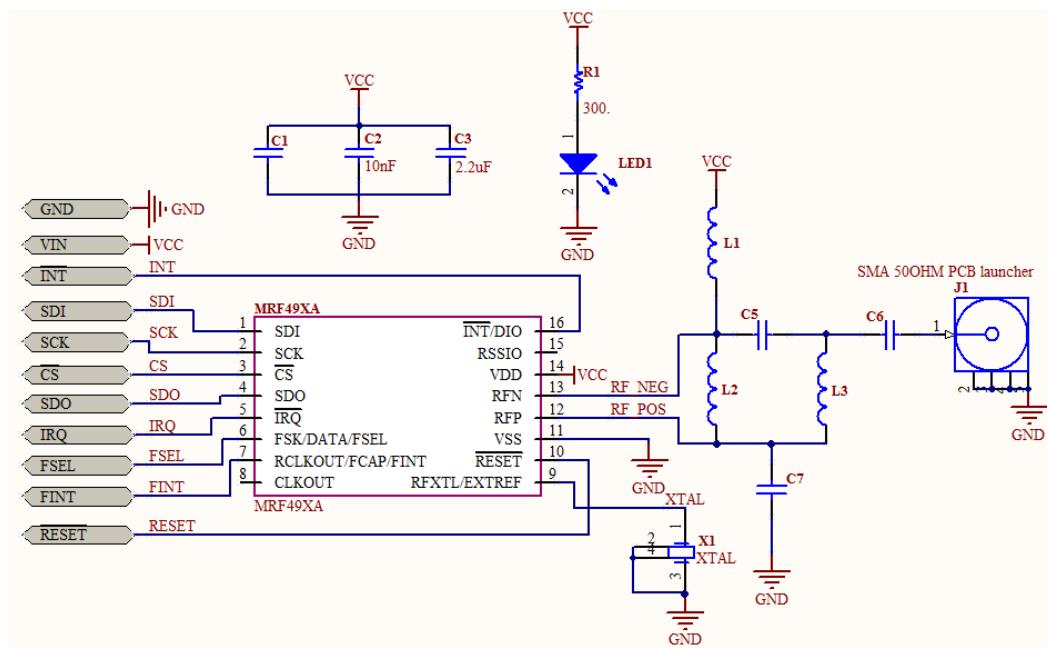


Figure A.1:  $\mu$ Trans 434/868 schematic.

Freq	C1	L1	L2	L3	C5	C6	C7
434 MHz	220 pF	390 nH	33 nH	47 nH	2.7pF	68 pF	5.1 pF
868 MHz	47 pF	100 nH	8.2 nH	22 nH	1.2 pF	27 pF	2.7 pF

Table A.1: Values for balun components at different frequency bands.



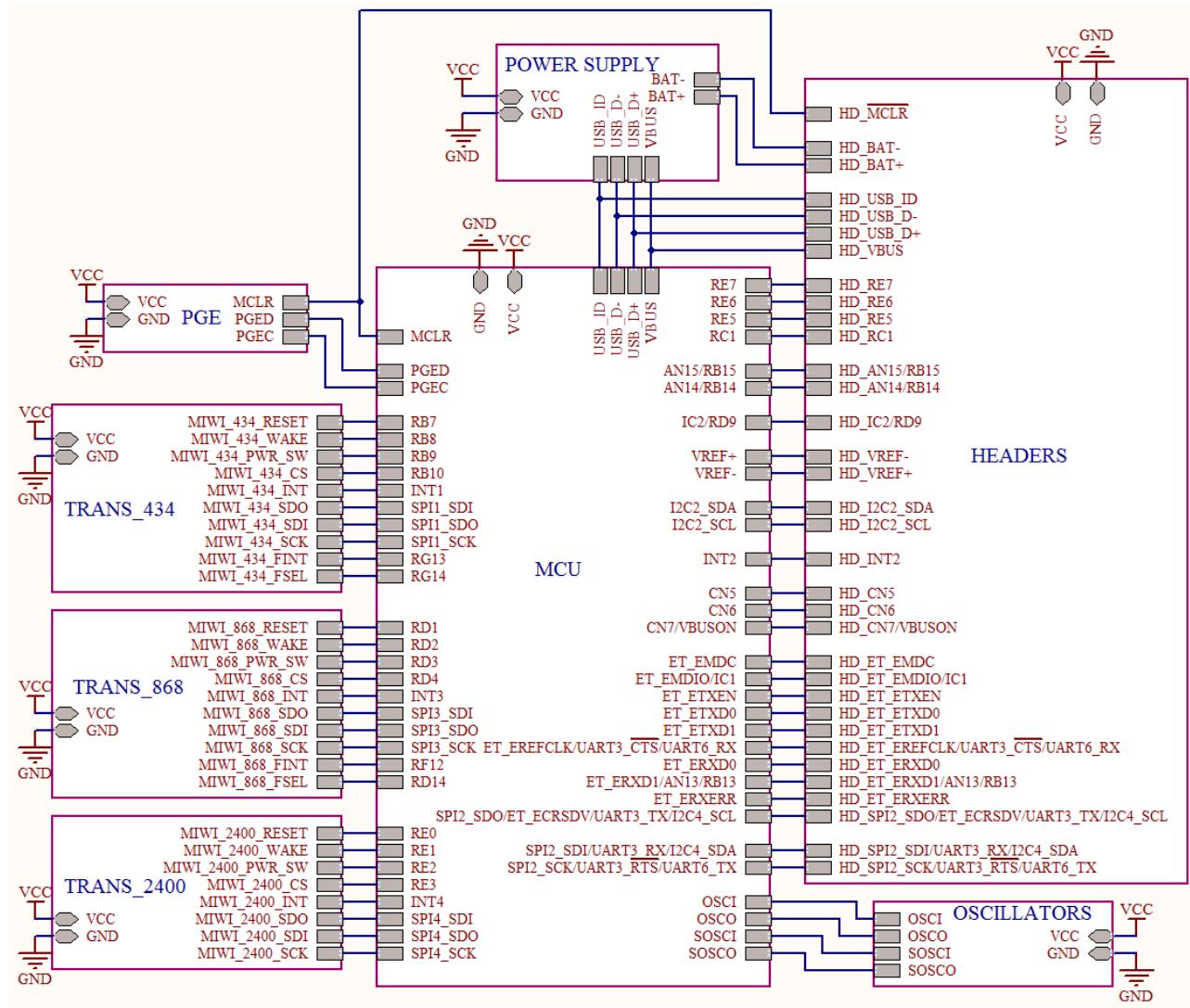


Figure A.2: cNGD general schematic.



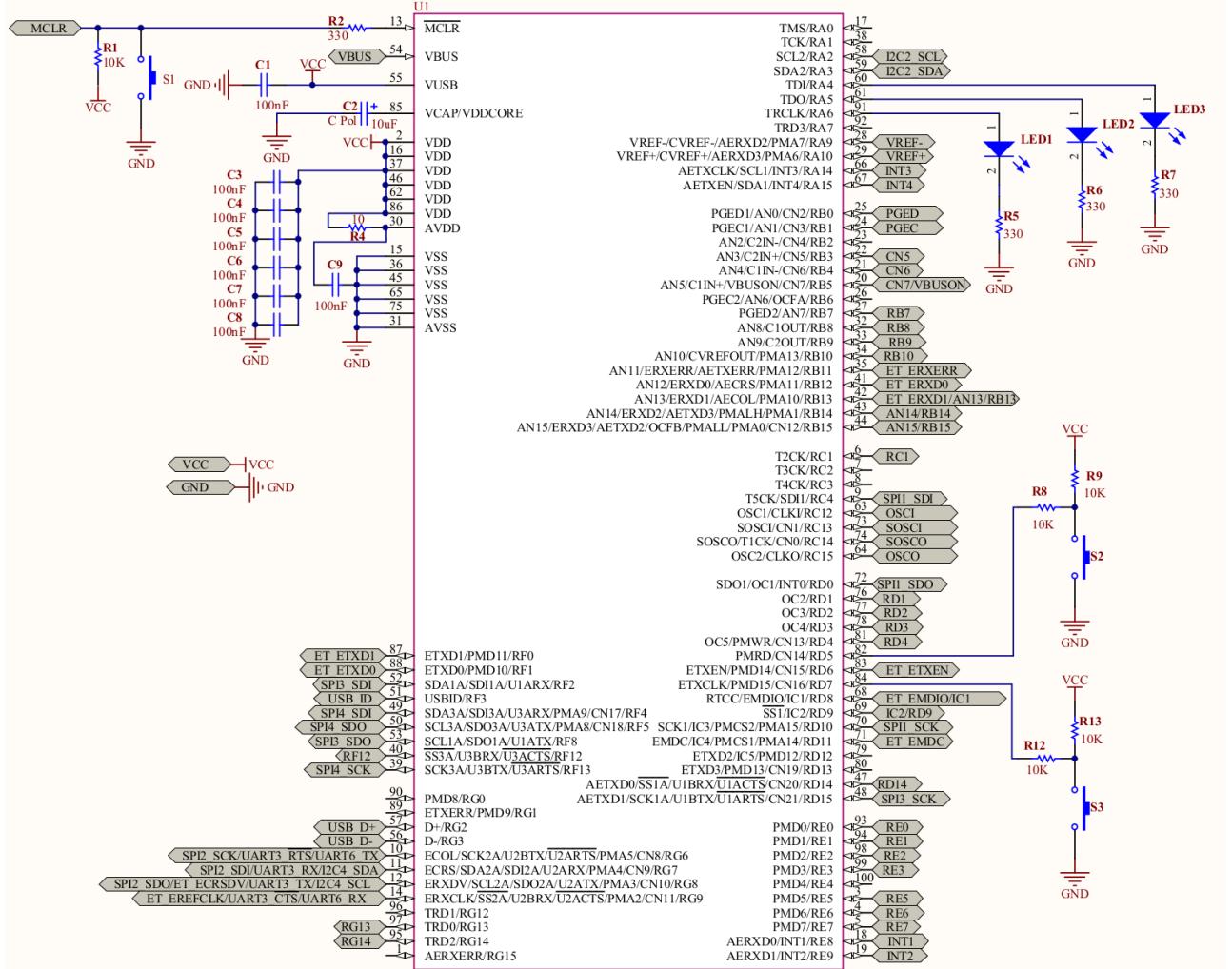


Figure A.3: Microcontroller specific schematic.



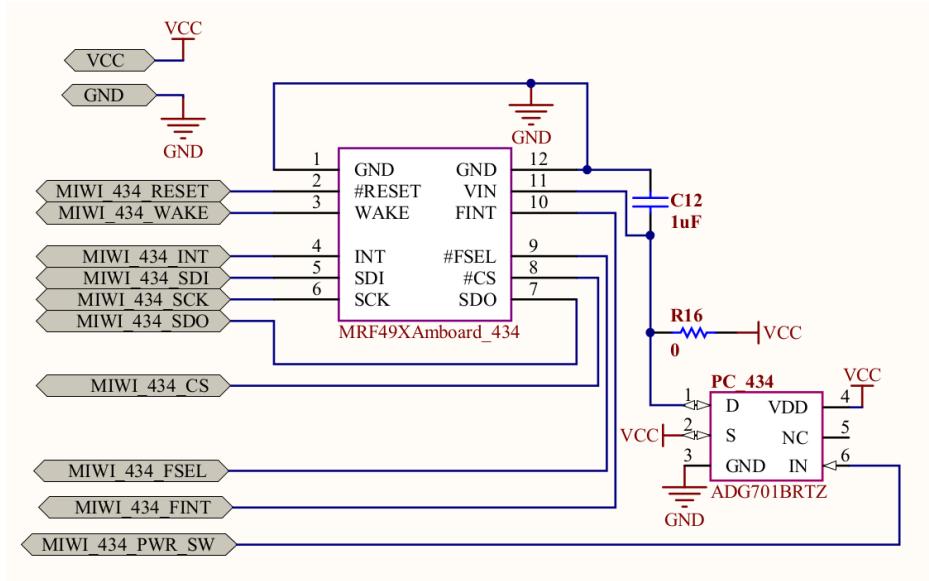


Figure A.4: 434 MHz Radio Interface systems schematic.

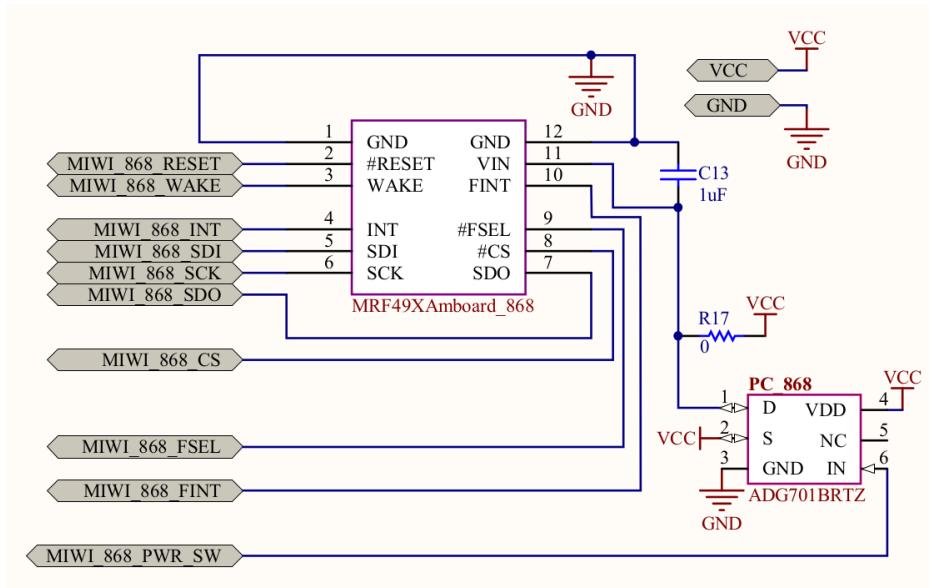
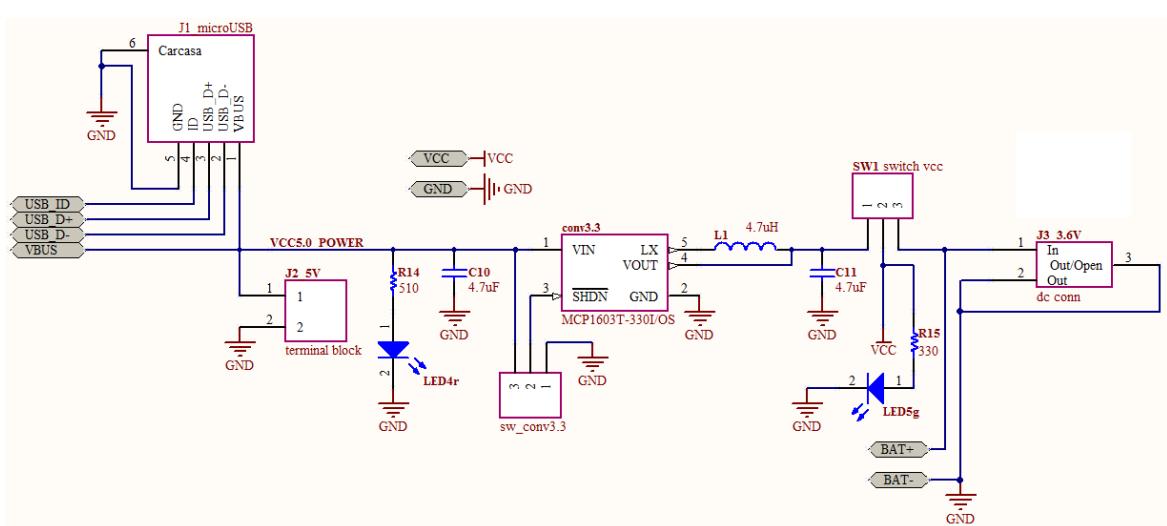
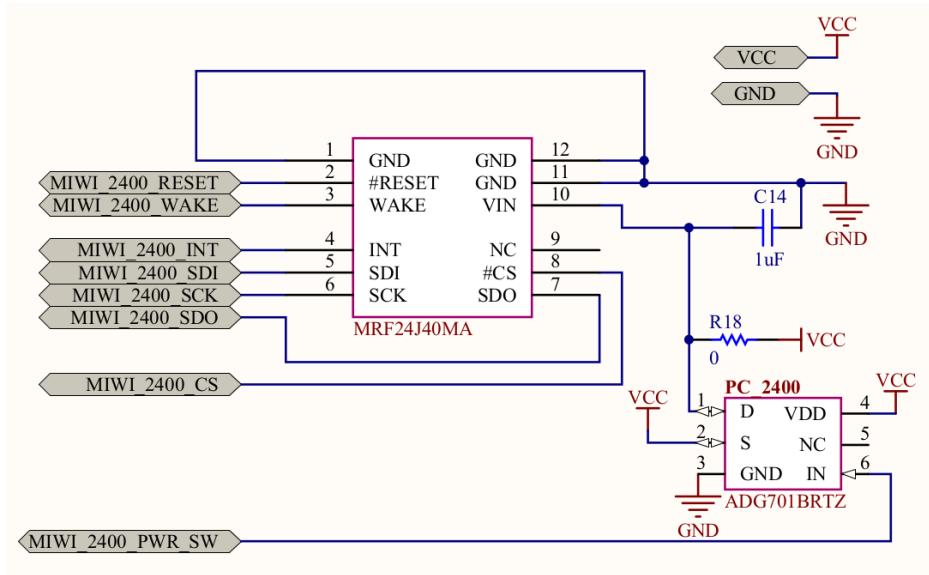


Figure A.5: 868 MHz Radio Interface systems schematic.







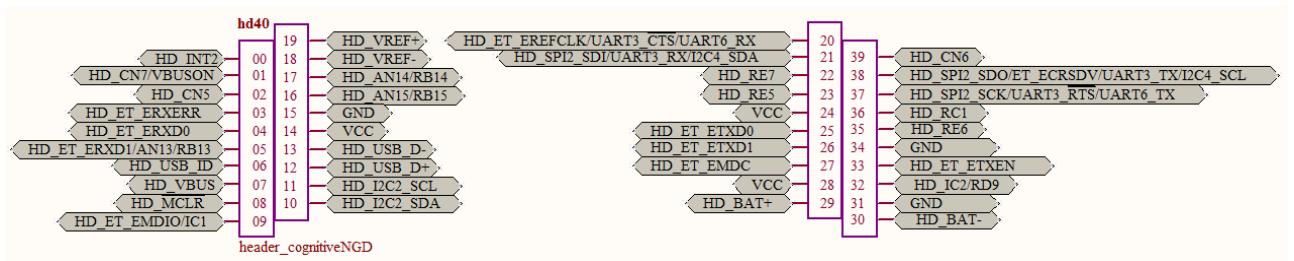


Figure A.8: Expansion header system schematic.

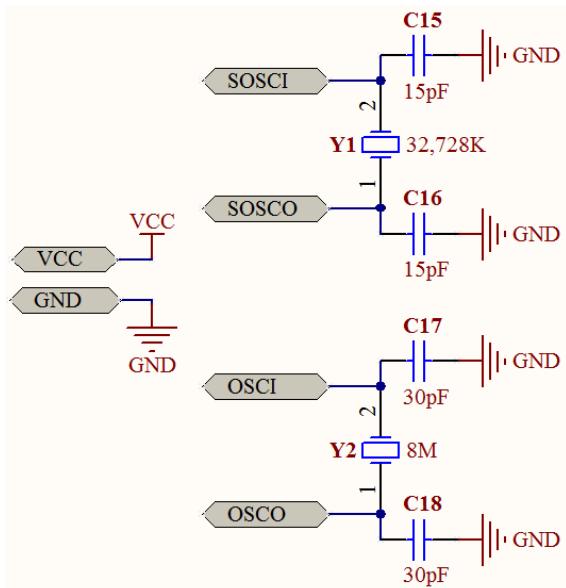


Figure A.9: Oscillator system schematic.

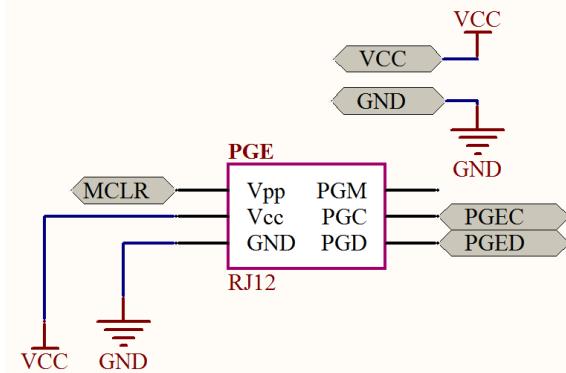


Figure A.10: PGE system schematic.



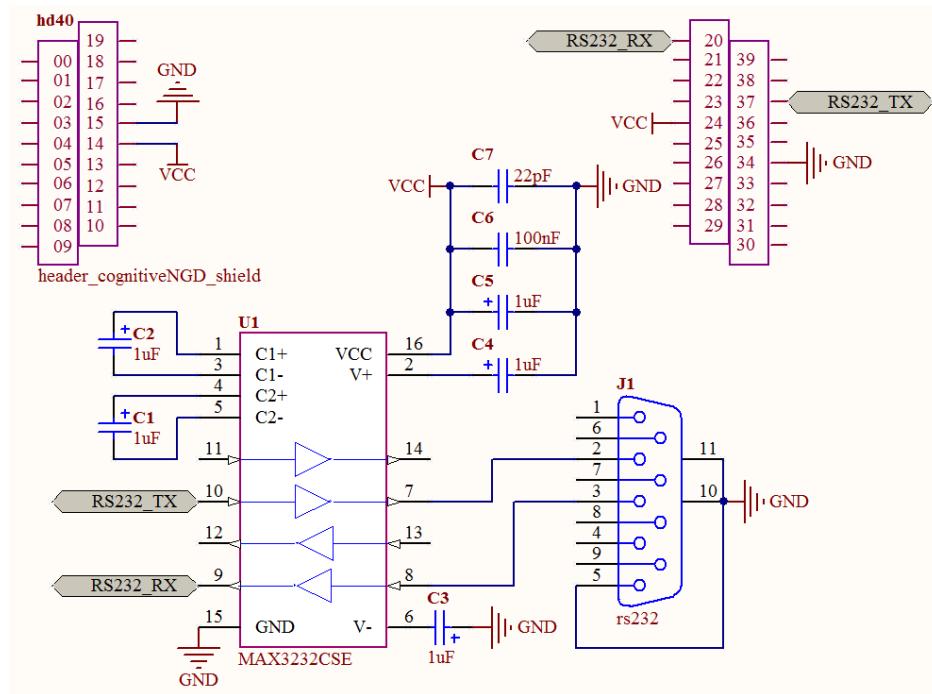


Figure A.11: rs232SHIELD schematic.

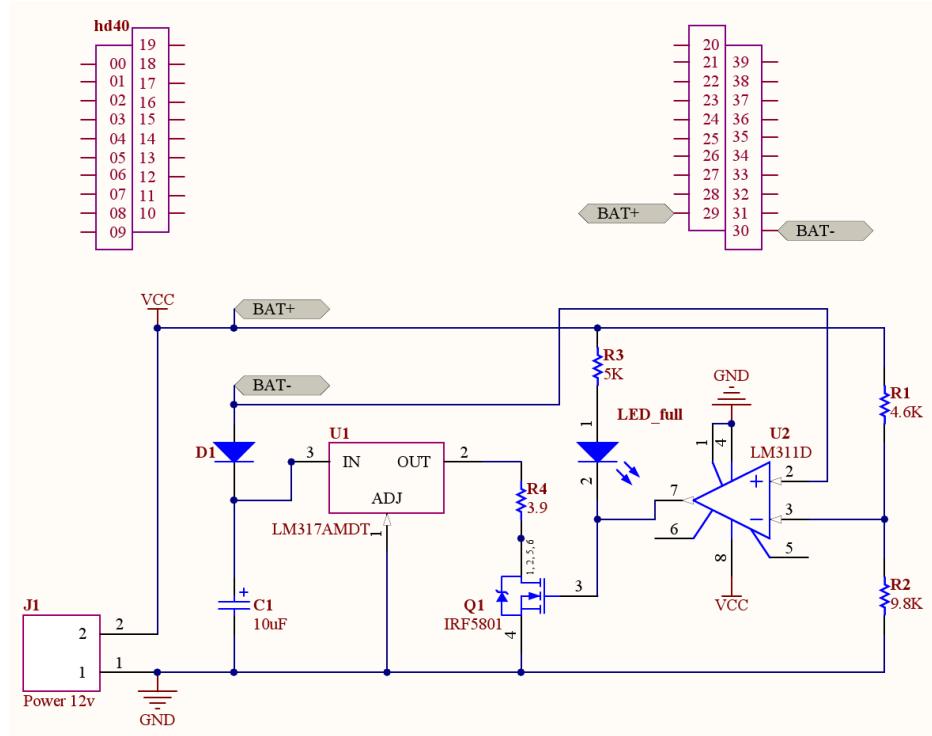


Figure A.12: chargerSHIELD schematic.



## Appendix B

# Layouts & Components

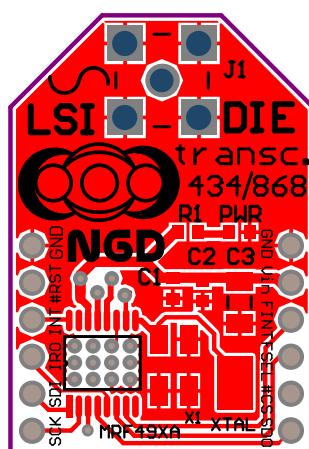


Figure B.1:  $\mu$ Trans Top layers.

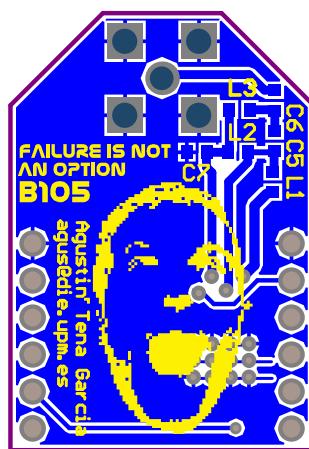


Figure B.2:  $\mu$ Trans Bottom layers.



Symbol	Nominal Value (434/868)	Description	Package Type	Quant.
C2	0.01 $\mu$ F	Ceramic Capacitor	0603	1
C3	2.2 $\mu$ F	Tantalum Capacitor	1206	1
X1	10MHZ	Quartz Crystal	5 x 3.2 mm SMD	1
MRF49XA		MRF49XA-I/ST - Transceiver, RF	16-TSSOP	1
J1		50 $\Omega$ Coaxial SMA Jack	Through Hole	1
D1		Light Emitting Diode	0603	1
R1	300 $\Omega$	Resistor	0603	1
C1	220 pF/47 pF	Ceramic Capacitor	0603	1
C5	2,7 pF/1,2 pF	Ceramic Capacitor	0603	1
C6	68 pF/27 pF	Ceramic Capacitor	0603	1
C7	5.1 pF/2,7 pF	Ceramic Capacitor	0603	1
L1	390 nH/100 nH	Inductor	0603	1
L2	33 nH/8.2 nH	Inductor	0603	1
L3	47 nH/22N nH	Inductor	0603	1
J1	434MHz/868MHZ	Monopole Antenna	SMA(M)	1

Table B.1: Component description for  $\mu$ Trans 434/868.



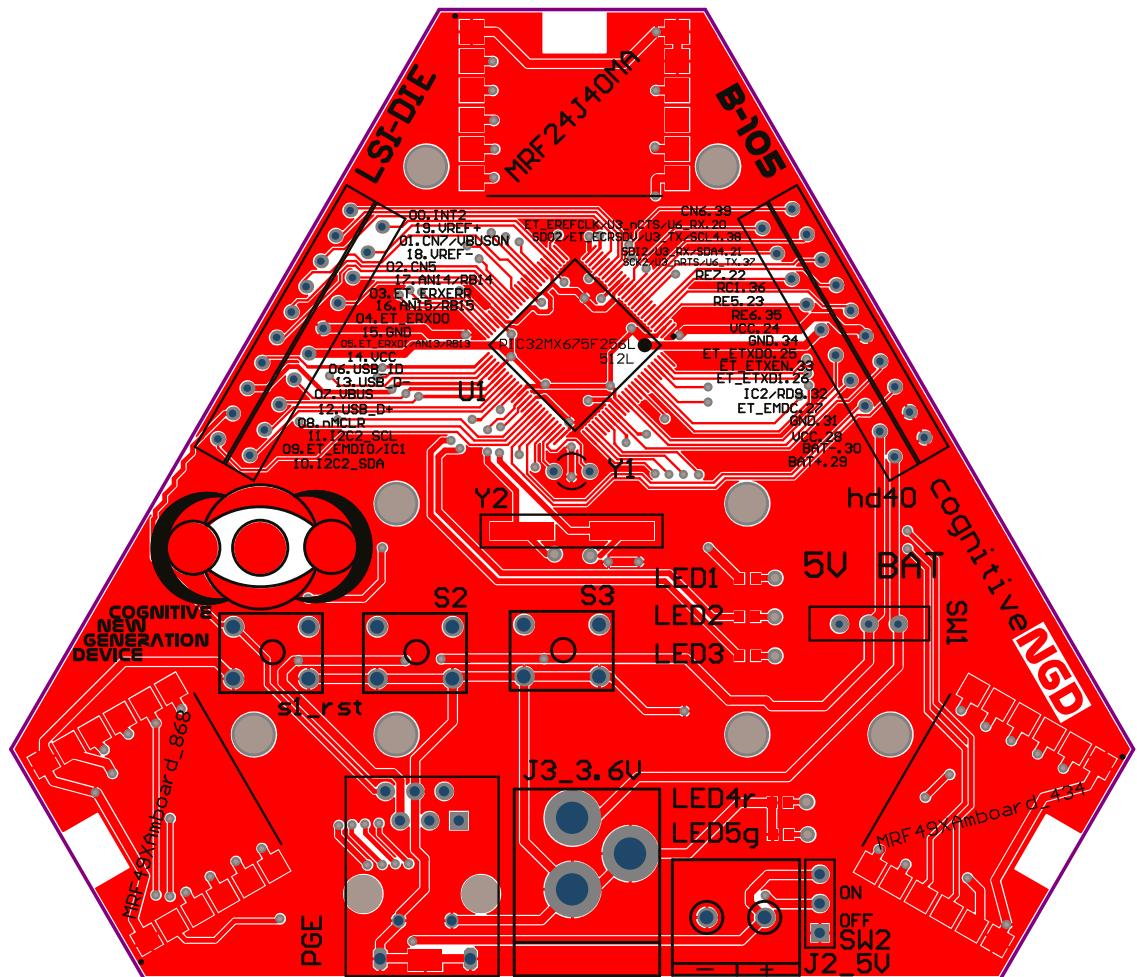


Figure B.3: cNGD Top layers.



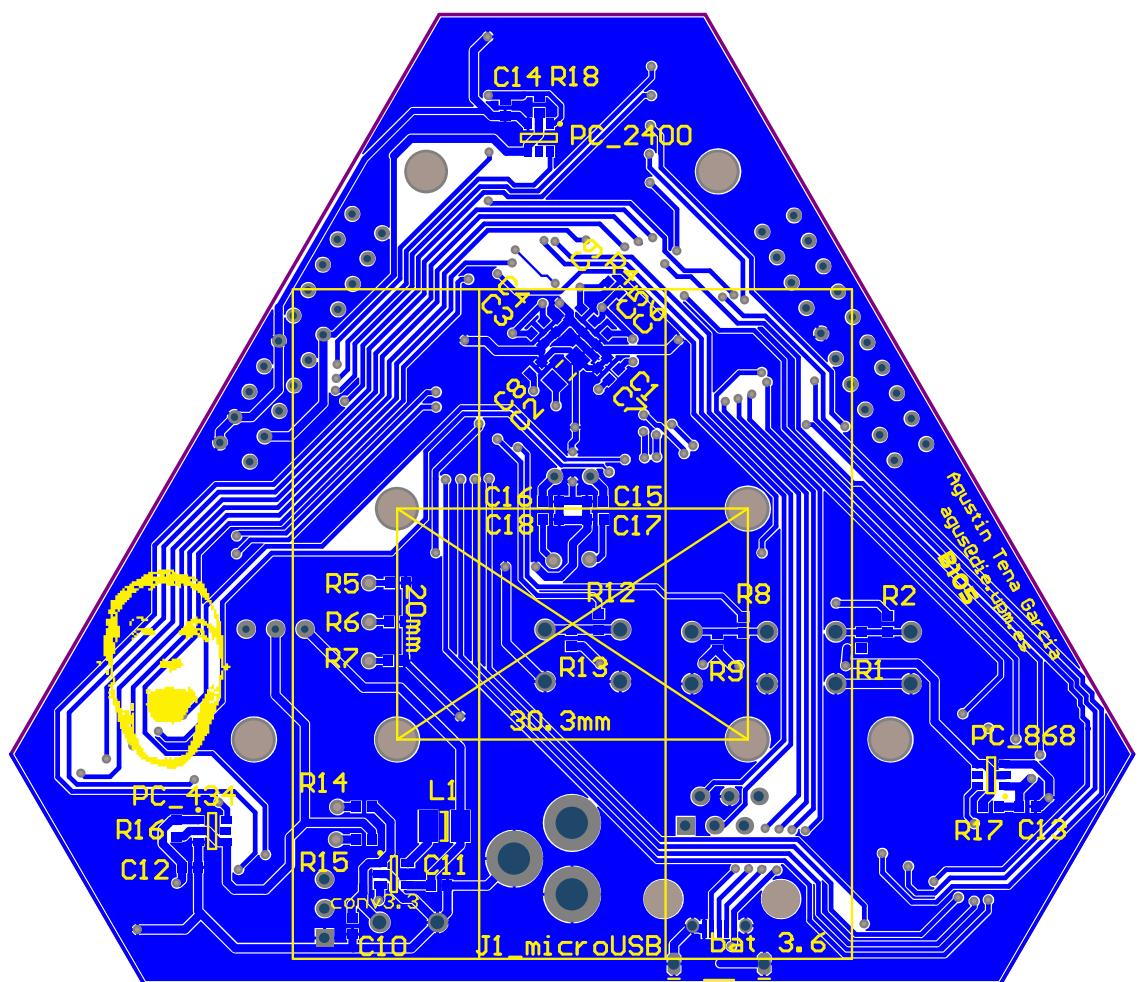


Figure B.4: cNGD Bottom layers.



Symbol	Nominal Value	Description	Package Type	Quant.
J1		MICRO USB A/B	SMD	1
C1,3-9	100 nF	Ceramic Capacitor	0603	8
C2	10 $\mu$ F	Pol. Capacitor	1206	1
C10,11	4.7 $\mu$ F	Ceramic Capacitor	0603	2
C12-14	1 $\mu$ F	Ceramic Capacitor	0603	3
C15,16	15 pF	Ceramic Capacitor	0603	2
C17,18	30 pF	Ceramic Capacitor	0603	2
R1,8,9,12,13	10 K $\Omega$	Resistor	0603	1
R4	10 $\Omega$	Resistor	0603	1
R2,5-7,15	330 $\Omega$	Resistor	0603	5
R15	510 $\Omega$	Resistor	0603	1
Y1	32.768 KHz	12.5pF Load capacitance crystal	Through hole	1
Y2	8 MHz	20 pF Load capacitance crystal	SMD	1
PGE		RJ-11 Connector		1
LED1-5		LED	0603	5
U1		PIC32MX675F256L-80I/PT	TQFP	1
hd40		10-pin Stackable Header		4
L1	4.7mH	Inductance	1210	1
PC_434, 868, 2400		ADG701LBRTZ - SWITCH	SOT-23	3
conv3.3		MCP1603T-330I/OS	TSOT	1
J2		Terminal Block		1
J3		DC Female Connector	Through hole	1
S1-3		Tactile Switch STSP	60x60 mm	3
sw1		Slide Switch, SPDT	Through Hole	1
bat3.6	3-cells	AA Batteries holder	Wired	1
		Legs	M3	4
		Screw	M3	4
		Nut	M3	4

Table B.2: Component description for cNGD.



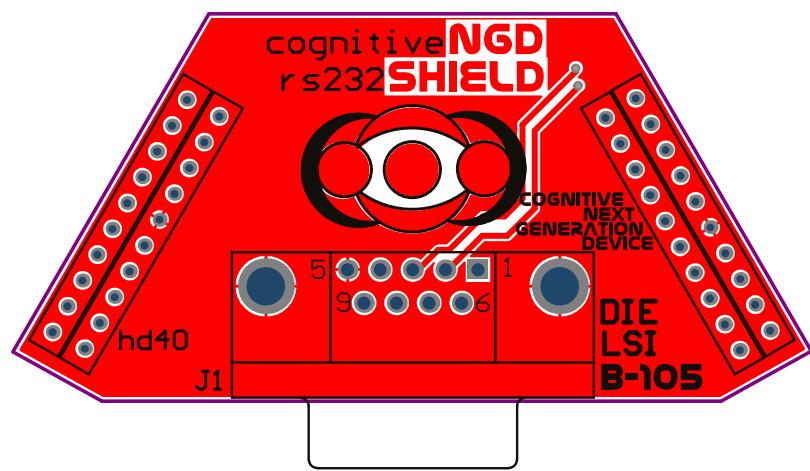


Figure B.5: rs232SHIELD Top layers.

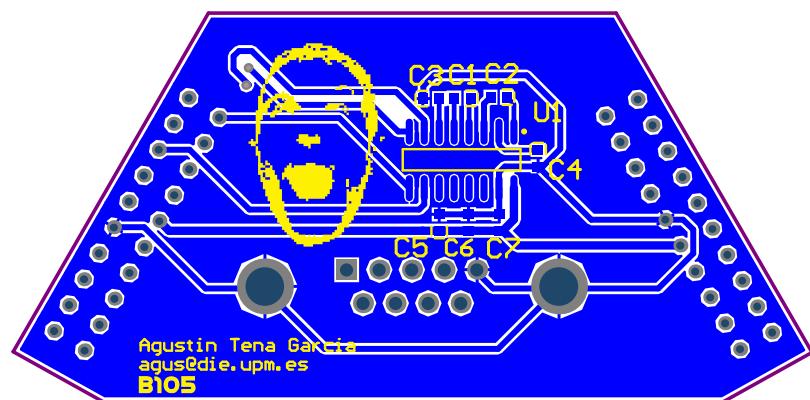


Figure B.6: rs232SHIELD Bottom layers.



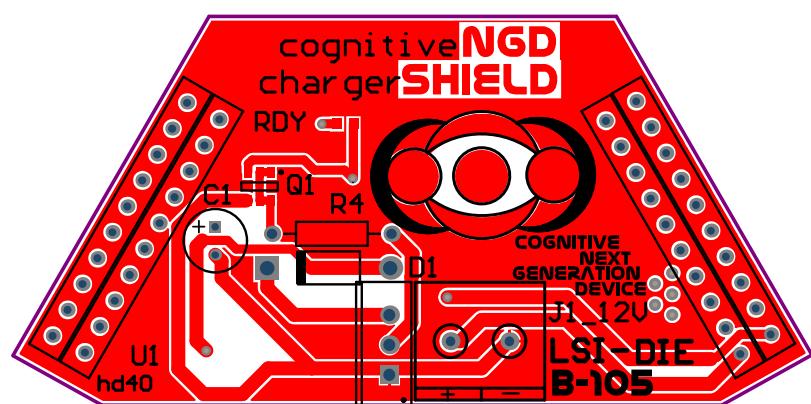


Figure B.7: chargerSHIELD Top layers.

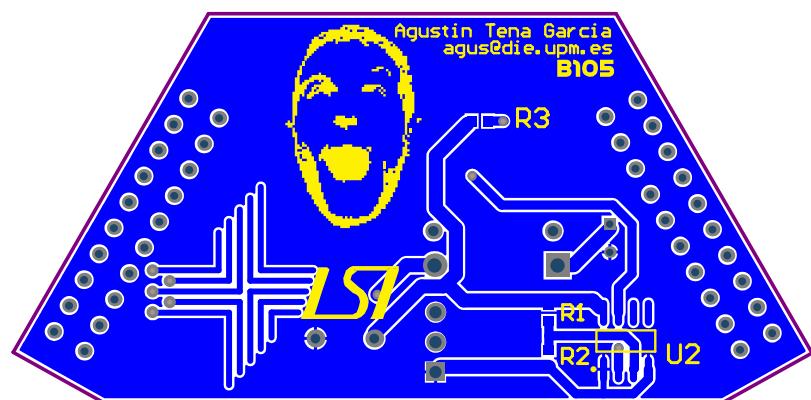


Figure B.8: chargerSHIELD Bottom layers.



Symbol	Nominal Value	Description	Package Type	Quant.
C1,2,3,4,5	1 $\mu$ F	Pol. Capacitor	0603	5
C6	100 nF	Ceramic Capacitor	0603	1
C7	22 pF	Ceramic Capacitor	0603	1
J1		DB9 female header - right angle	PCB mount	1
U1		MAX3232CD - RS232 TXRX	SOIC	1
hd40		10-pin Stackable Header		4

Table B.3: Components description for rs232SHIELD.

Symbol	Nominal Value	Description	Package Type	Quant.
J1		Terminal Block		1
R3	5K	Resistor	0603	1
R4	3.9 $\Omega$	1W Resistor	Axial	1
C1	10 $\mu$ F	Terminal Block		1
U1		LM317AMDT - IC, V. REG.	TO-220	1
hd40		10-pin Stackable Header		4
Q1		MOSFET IRF5801	TSOP	1
R1	4.6 K $\Omega$	Resistor	0603	1
R2	9.8 K $\Omega$	Resistor	0603	1
U2		LM311D - IC, COMPARATOR	SOIC8	1
D1		DIODE, STANDARD, 0.5A	Axial Leaded	1

Table B.4: Components description for chargerSHIELD.



# References

- [1] Perera, C.; Zaslavsky, A.; Christen, P.; Georgakopoulos, D., “Context Aware Computing for The Internet of Things: A Survey,” *Communications Surveys & Tutorials, IEEE*, vol. abs/1305.0982, pp. 1 – 4, May 2013.
- [2] Howitt, I.; Gutierrez, J.A., “IEEE 802.15.4 low rate - wireless personal area network coexistence issues,” *Wireless Communications and Networking, WCNC 2003*, vol. 3, pp. 1481 – 1486, Mar 2003.
- [3] “MapAbility for Map Resources and Overlay Mapping.” <http://www.mapability.com/>. Last accessed on 28-10-2013.
- [4] Mitola, J. III; Maguire, G.Q. Jr, “Cognitive radio: making software radios more personal,” *Personal Communications*, vol. 6, pp. 13 – 18, Aug 1999.
- [5] Mitola, J. III, *Cognitive Radio An Integrated Agent Architecture for Software Defined Radio, PhD. Dissertation*. Royal Institute of Technology, Kista, Sweden, May 2000.
- [6] Akan, O.B.; Karli, O.; Ergul, O., “Cognitive radio sensor networks,” *Network, IEEE*, vol. 23, pp. 34 – 40, Jul 2009.
- [7] Vijay, G.; Ben Ali Bdira, E.; Ibnkahla, M., “Cognition in wireless sensor networks: A perspective,” *IEEE Sensors Journal*, vol. 11, pp. 582 – 592, Mar 2011.
- [8] López, F.; Romero, E.; Blesa, J.; Villanueva, D.; Araujo, A., “Cognitive Wireless Sensor Network Device for AAL Scenarios,” *Lecture Notes in Computer Science*, vol. 6693, pp. 116 – 121, 2011.
- [9] Domingo, J., *Diseño, optimización y prueba un nodo para una red de sensores inalámbrica con capacidades cognitivas, PFC*. Laboratorio de Sistemas Integrados, ETSIT-UPM, Feb 2013.
- [10] Jara, G., *Diseño e implementación de una arquitectura para la gestión de comunicaciones de una red de sensores inalámbricas cognitiva*. Laboratorio de Sistemas Integrados, ETSIT-UPM, Sep 2013.
- [11] Rabaey, J.; Wolisz, A.; Ozer Ercan, A.; Araujo, A.; Burghardt, F.; Mustafa, S.; Parsa, A.; Pollin, S.; I-Hsiang Wang; Malagon P., “Connectivity Brokerage - Enabling Seamless Cooperation in Wireless Networks,” tech. rep., Berkeley Wireless Research Center, Sep 2010.
- [12] “Latex.” <http://en.wikibooks.org/wiki/LaTeX/>. Last accessed on 11-9-2013.

- [13] Brodersen, R.W.; Wolisz, A.; Cabric, D.; Mishra, S.M.; Willkomm, D., “Corvus: A cognitive radio approach for usage of virtual unlicensed spectrum,” 2004.
- [14] Federal Communications Commission, “Spectrum Policy Task Force,” *Rep. ET Docket*, vol. 2, p. 135, Nov 2002.
- [15] Jong Kim, S., “Dynamic Spectrum Allocation with Variable Bandwidth for Cognitive Radio Systems,” *Communications and Information Technology*, pp. 106 – 109, Sep 2009.
- [16] Haykin, S., “Cognitive radio: Brain-empowered wireless communications,” *Selected Areas in Communications, IEEE Journal on*, vol. 23, pp. 201 – 220, Feb 2005.
- [17] Federal Communications Commission, “Facilitating opportunities for flexible, efficient, and reliable spectrum use employing cognitive radio technologies, notice of proposed rulemaking and order, fcc 03-322,” Dec 2003.
- [18] Ying-Chang Liang, Kwang-Cheng Chen ; Li, G.Y. ; Mahonen, P., “Cognitive radio networking and communications: An overview,” *Vehicular Technology, IEEE Transactions on*, vol. 60, pp. 3386 – 3407, Jun 2011.
- [19] “IEEE 802.15 WPAN<sup>TM</sup> Task Group 2.” <http://www.ieee802.org/15/pub/TG2.html>. Last accessed on 12-9-2013.
- [20] Akyildiz, I.F.; Won-Yeol Lee; Vuran, M.C.; Mohanty, S., “NeXt generation/dynamic spectrum access/cognitive Radio Wireless Networks: A Survey,” *Computer Networks (Elsevier) Journal*, vol. 50, pp. 2127 – 2159, Sep 2006.
- [21] Venkatesha, R.; Hoffmeyer, J.A.; Berger, H.S., “Cognitive functionality in next generation wireless networks,” *IEEE Communications Magazine*, pp. 72 – 78, Apr 2008.
- [22] Zhiqiang Li; F. Richard Yu; Minyi Huang, “A distributed consensus-based cooperative spectrum sensing in cognitive radios,” *IEEE Trans. Vehicular Technology*, vol. 59, pp. 383 – 393, Jan 2010.
- [23] Foukalas, F.T.; Mathiopoulos, P.T.; Karetos, G.T., “Joint optimal power allocation and sensing threshold selection for SU’s capacity maximisation in SS CRNS,” *Electronics Letters*, vol. 46, pp. 1406 – 1407, Sep 2010.
- [24] “IEEE 802.22 Working Group on Wireless Regional Area Networks.” <http://www.ieee802.org/22/>. Last accessed on 12-9-2013.
- [25] Stevenson, C.; Chouinard, G.; Zhongding Lei; Wendong Hu; Shellhammer, S.J.; Caldwell, W., “Ieee 802.22: The first cognitive radio wireless regional area network standard,” *Communications Magazine, IEEE*, vol. 47, pp. 130 – 138, Jan 2009.
- [26] Alexiou, A. ; Haardt, M., “Smart antenna technologies for future wireless systems: Trends and challenges,” *IEEE Communications Magazine*, vol. 42, pp. 90 – 97, Sep 2004.
- [27] Berezdivin, R.; Breinig, R.; Topp, R., “Next-generation wireless communications concepts and technologies,” *IEEE Communications Magazine*.

- [28] Schwartz, B.; Jackson, A.W.; Strayer, W.T.; Wenyi Zhou; Rockwell, R.D.; Partridge, C., "Smart packets for active networks," *OPENARCH '99*, pp. 90 – 97, 1999.
- [29] Gelenbe, E.; Lent, R.; Zhiguang Xu, "Design and performance of cognitive packet networks," *Performance Evaluation*, vol. 46, pp. 155 – 176, Mar 2001.
- [30] Mähönen, P.; Riihijärvi, J.; Petrova, M.; Shelby, Z., "Hop-by-hop toward future mobile broadband IP," *IEEE Communications Magazine*, vol. 42, pp. 138 – 146, Mar 2004.
- [31] Ramming, C., "Cognitive networks," in *DARPAtech*, Mar 2004.
- [32] Saracco, R., "Forecasting the future of information technology: How to make research investment more cost-effective," *IEEE Communications Magazine*, vol. 41, pp. 38 – 45, Dec 2003.
- [33] Neel, J.O.; Reed, J.H.; Gilles, R.P., "Convergence of cognitive radio networks," in *Wireless Communications and Networking Conference*, vol. 4, pp. 2250 – 2255, 2004.
- [34] Neel, J.O.; Buehrer, R.M; Reed, J.H.; Gilles, R.P., "Game-theoretic analysis of a network of cognitive radios," in *Circuits and Systems, 45th Midwest Symposium on*, vol. 3, pp. 409 – 412, 2002.
- [35] Ganesan, G.; Ye Li, "Cooperative spectrum sensing in cognitive radio networks," in *New Frontiers in Dynamic Spectrum Access Networks. First IEEE International Symposium on*, pp. 137 – 143, 2005.
- [36] Mishra, S.M.; Sahai, A.; Brodersen, R.W., "Cooperative sensing among cognitive radios," in *Communications. IEEE International Conference on*, vol. 4, pp. 1658 – 1663, 2006.
- [37] Pawelczak, P.; Prasad, R.V.; Liang Xi; Niemegeers, I.G.M.M., "Cognitive radio emergency networks - requirements and design," in *New Frontiers in Dynamic Spectrum Access Networks. First IEEE International Symposium on*, pp. 601 – 606, 2005.
- [38] Raychaudhuri, D.; Mandayam, N.B; Evans, J.B.; Ewy, B.J.; Seshan, S.; Steenkiste, P., "CogNet: An architectural foundation for experimental cognitive radio networks within the future Internet," *Proc. ACM/IEEE MobiArch'06*, pp. 11 – 16, Dec 2006.
- [39] Clark, D. D., Partridge, C., Ramming, J. C., and Wroclawski, J. T., "A knowledge plane for the internet," *Computer Communication Review*, vol. 33, pp. 3 – 10, Oct 2003.
- [40] Thomas, R.W., *Cognitive Networks, PhD. Dissertation*. Faculty of the Virginia Polytechnic Institute and State University, Blacksburg, Virginia, Jun 2007.
- [41] Thomas, R.W.; Friend, D.H.; Da Silva, L.A.; MacKenzie A.B., "Cognitive networks: Adaptation and learning to achieve end-to-end performance objectives," *IEEE Communications Magazine*, vol. 44, pp. 51 – 57, Dec 2006.

- [42] Utrilla, R., *Diseño y realización de un prototipo de red ad hoc de datos con aplicación en la monitorización de cetáceos, PFC*. Laboratorio de Aplicaciones Bioacústicas, UPC, Sep 2011.
- [43] “IEEE 802.15. WPANTM Task Group 4.” <http://grouper.ieee.org/groups/802/15/pub/TG4.html>. Last accessed on 12-8-2013.
- [44] “Zigbee Alliance.” [www.zigbee.org](http://www.zigbee.org). Last accessed on 11-9-2013.
- [45] “HART Communication Foundation, WirelessHART Technology.” <http://www.hartcomm.org>. Last accessed on 11-9-2013.
- [46] “ISA, Wireless Compliance Institute. ISA100 Wireless Systems for Automation.” <http://www.isa100wci.org>. Last accessed on 11-9-2013.
- [47] Microchip Technology Inc., *AN1066 Microchip MiWiTM Wireless Networking Protocol Stack*. 2007.
- [48] “IEEE 802.11TM Wireless Local Area Networks.” <http://www.ieee802.org/11/>. Last accessed on 11-9-2013.
- [49] “Wi-Fi Alliance.” <http://www.wi-fi.org>. Last accessed on 11-9-2013.
- [50] “IETF. IPv6 over Low power WPAN.” <http://datatracker.ietf.org/wg/6lowpan/>. Last accessed on 11-9-2013.
- [51] Machado, R.; Tekinay, S., “A survey of game-theoretic approaches in wireless sensor networks,” *Computer Network*, vol. 52, pp. 3047 – 3061, Nov 2008.
- [52] Reznik, L.; Von Pless, G., “Neural networks for cognitive sensor networks,” *Proc. IEEE Int. Joint Conf. Neural Network., IJCNN 2008*, pp. 1235 – 1241, Jun 2008.
- [53] Akan, O.B.; Karli, O.; Ergul, O., “Cognitive radio sensor networks,” *IEEE Network*, vol. 23, pp. 34 – 40, Jul 2009.
- [54] Cavalcanti, D.; Das, S.; Jianfeng Wang; Challapali, K., “Cognitive radio based wireless sensor networks,” *Proc. 17th Int. Conf. Comput. Commun. Networks, ICCCN*, pp. 1 – 6, 2008.
- [55] “TelosB.” <http://www.advanticsys.com/shop/mtmcm5000msp-p-14.html>. Last accessed on 11-9-2013.
- [56] ANT platform. <http://www.thisisant.com>. Last accessed on 11-9-2013.
- [57] Eyes Project. <http://www.eyes.eu.org>. Last accessed on 11-9-2013.
- [58] Mishra, S.M.; Cabric, D.; Chang, C.; Willkomm, D.; van Schewick, B.; Wolisz, A.; Brodersen, R.W., “A real time cognitive radio testbed for physical and link layer experiments,” *Proceedings of the 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, vol. 1, pp. 562 – 567, Nov 2005.
- [59] “OpenAirInterface.” <http://www.openairinterface.org>. Last accessed on 11-9-2013.

- [60] Harada, H., "Software defined radio prototype toward cognitive radio communication systems," *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, pp. 539 – 547, Nov 2005.
- [61] Minden, G.J.; Evans, J.B.; Searl, L.; DePardo, D.; Petty, V.R.; Rajbanshi, R.; Newman, T.; Chen, Q.; Weidling, F.; Guffey, J.; Datla, D.; Barker, B.; Peck, M.; Cordill, B.; Wyglinski, A.M.; Agah, A., "Kuar: A flexible software-defined radio development platform," *New Frontiers in Dynamic Spectrum Access Networks, 2007. DySPAN 2007. 2nd IEEE International Symposium on*, pp. 428 – 439, Apr 2007.
- [62] Berardinelli, G; Zetterberg, P.; Tonelli, O.; Cattoni, A.F.; Sorensen, T.B.; Mogensen, P., "An SDR architecture for OFDM transmission over USRP2 boards," *Signals, Systems and Computers (ASILOMAR), 2011 Conference Record of the Forty Fifth Asilomar Conference on*, pp. 965 – 969, Nov 2011.
- [63] Yunjiao Xue; Ho Sung Lee; Ming Yang; Kumarawadu, P.; Ghenniwa, H.H.; Weiming Shen, "Performance evaluation of NS-2 simulator for wireless sensor networks," *Electrical and Computer Engineering, 2007. CCECE 2007. Canadian Conference on*, pp. 1372 – 1375, Apr 2007.
- [64] Myungin Ji; Yonggyu Kim; Myunghwan Seo; Joongsoo Ma, "A novel multi-channel multi-radio wireless mesh node architecture for NS-2," *Communications and Mobile Computing, 2009. CMC '09. WRI International Conference on*, vol. 2, pp. 147 – 151, Jan 2009.
- [65] Pongor, G., "Omnet: Objective modular network testbed," *MASCOTS '93 Proceedings of the International Workshop on Modeling, Analysis, and Simulation On Computer and Telecommunication Systems*, pp. 323 – 326, 1993.
- [66] MiXiM Project. <http://mixim.sourceforge.net/>. Last accessed on 23-9-2013.
- [67] "Universität Paderborn. Wireless channel simulator for OMNeT++." <http://www.cs.uni-paderborn.de/en/fachgebiete/research-group-computer-networks/projects/chsim.html>. Last accessed on 23-9-2013.
- [68] "Mobility Framework for OMNeT++." <http://mobility-fw.sourceforge.net/>. Last accessed on 23-9-2013.
- [69] "Consensus: Collaborative Sensor Networks." <http://www.consensus.tudelft.nl/software.html>. Last accessed on 23-9-2013.
- [70] "Castalia. Wireless Sensor Network Simulator." <http://castalia.research.nicta.com.au>. Last accessed on 11-9-2013.
- [71] Araujo, A.; Romero, E.; Blesa, J.; Nieto-Taladriz, O., "A framework for the design, development and evaluation of cognitive wireless sensor networks," *International Journal On Advances in Telecommunications*, vol. 5, pp. 141 – 152, Dic 2012.
- [72] "NetSim Cisco Network Simulator and Router Simulator." <http://www.boson.com/netsim-cisco-network-simulator>. Last accessed on 11-9-2013.

- [73] Pescosolido, L., “Performance of SENDORA networks,” 2011.
- [74] López, F., *Diseño e implementación de un nodo para redes de sensores con capacidades cognitivas, PFC*. Laboratorio de Sistemas Integrados, ETSIT-UPM, Dic 2011.
- [75] “SensorLab. Hardware. VESNA.” <http://sensorlab.ijs.si/hardware.html>. Last accessed on 12-9-2013.
- [76] “SensorLab.” <http://sensorlab.ijs.si/>. Last accessed on 23-9-2013.
- [77] Solč, T., “SNE-ISMTV: VESNA wireless sensor node expansion for cognitive radio experiments,” *ISWCSC 2013 - The Tenth International Symposium on Wireless Communication Systems*, p. 2, Aug 2013.
- [78] Home page - FP7 - Research - Europa. <http://ec.europa.eu/research/fp7/>. Last accessed on 25-9-2013.
- [79] COGEU. <http://www.ict-cogeu.eu/>. Last accessed on 25-9-2013.
- [80] Project Overview. CREW project. <http://www.crew-project.eu/>. Last accessed on 25-9-2013.
- [81] Handziski, V.; Köpke, A.; Willig, A.; Wolisz, A., “Twist: a scalable and reconfigurable testbed for wireless indoor experiments with sensor networks,” *REALMAN '06 Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality*, pp. 63 – 70, 2006.
- [82] Moteiv Cormporation, “Tmote Sky datasheet,” Jun 2006.
- [83] “LOG-a-TEC by SensorLab.” <http://log-a-tec.eu/>. Last accessed on 28-10-2013.
- [84] The iMinds Organization - iMinds. <http://www.iminds.be/en>. Last accessed on 25-9-2013.
- [85] “PC Engines alix3d3 product file.” <http://pcengines.ch/alix3d3.htm>. Last accessed on 11-9-2013.
- [86] Newman, T.R.; He, A.; Gaeddert, J.; Hilburn, B.; Bose, T.; Reed, J.H., “Virginia tech cognitive radio network testbed and open source cognitive radio framework,” *Testbeds and Research Infrastructures for the Development of Networks & Communities and Workshops, 2009. TridentCom 2009. 5th International Conference on*, pp. 1 – 3, Apr 2009.
- [87] “FIT/CortexLab: Cognitive radio testbed.” <http://www.cortexlab.fr/>. Last accessed on 12-9-2013.
- [88] “FIT: Future Internet (of Things).” <http://fit-equipex.fr/english>. Last accessed on 12-9-2013.
- [89] “Senslab - Very large scale open wireless sensor network testbed.” <http://www.senslab.info/>. Last accessed on 23-9-2013.
- [90] Microchip Technology Inc., *PIC32MX5XX/6XX/7XX Family Data Sheet*. 2009.

- [91] Advanced Wireless Dynamics, *Tulio<sup>TM</sup> Datasheet*. 2011.
- [92] Microchip Technology Inc., *MRF24J40MA Data Sheet 2.4 GHz, IEEE Std. 802.15.4<sup>TM</sup> RF Transceiver Module*. 2008.
- [93] Microchip Technology Inc., *AN1204 Microchip MiWi<sup>TM</sup> P2P Wireless Protocol*. 2010.
- [94] Microchip Technology Inc., *AN1371 Microchip MiWi<sup>TM</sup> PRO Wireless Networking Protocol*. 2011.
- [95] Microchip Technology Inc., *Microchip MRF24WB0MA/MRF24WB0MB Data Sheet 2.4 GHz, IEEE Std. 802.11b<sup>TM</sup> RF Transceiver Module*. 2010.
- [96] Microchip Technology Inc., *AN833 The Microchip TCP/IP Stack*. 2008.
- [97] Microchip Technology Inc., *MCP1612 Single 1A, 1.4 MHz Synchronous Buck Regulator*. 2002.
- [98] Microchip Technology Inc., *MCP1252/3 Low Noise, Positive-Regulated Charge Pump*.
- [99] Microchip Technology Inc., *MRF49XA PICtail<sup>TM</sup>/PICtail Plus Daughter Board User's Guide*. 2009.
- [100] Microchip Technology Inc., *MRF49XA Data Sheet, ISM Band Sub-GHz RF Transceiver*. 2009.
- [101] Microchip Technology Inc. <http://www.microchip.com>. Last accessed on 28-10-2013.
- [102] Analog Devices, *ADG701/ADG702. CMOS Low Voltage 2 Ω SPST Switches. Datasheet*. 2006.
- [103] Microchip Technology Inc., *MCP1603/B/L 2.0 MHz, 500 mA Synchronous Buck Regulator*.
- [104] FOX, *HC49SDLF Resistance Weld SMD Crystal*. 2008.
- [105] ABRACON Corporation, *AB15T Low frequency cylindrical watch crystal*. 2008.
- [106] Texas Instrument, *MAX3232 Datasheet*. 2004.
- [107] Texas Instrument, *LM111, LM211, LM311 DIFFERENTIAL COMPARATORS WITH STROBES*. 2003.
- [108] Texas Instrument, *LM117, LM317A, LM317-N Datasheet*. 2013.
- [109] International Rectifier, *IRF5801 SMPS MOSFET*. 2002.
- [110] “JBC. Rework Station 6000. Instructions manual.” [http://www.jbctools.com/pdf/manual\\_AM6000\\_230v.pdf](http://www.jbctools.com/pdf/manual_AM6000_230v.pdf). Last accessed on 16-9-2013.
- [111] Microchip Technology Inc., *MPLAB ICD 3 In-Circuit Debugger User's Guide For MPLAB X IDE*. 2010.

- [112] Microchip Technology Inc., *Microchip USB Device Firmware Framework User's guide*. 2008.
- [113] Microchip Technology Inc., *AN1166 USB Generic Function on an Embedded Device*. 2008.
- [114] Microchip Technology Inc., *AN1164 USB CDC Class on an Embedded Device*. 2008.
- [115] Rodríguez, R., *Diseño e implementación de un módulo de activación por radio frecuencia para redes de sensores inalámbricas, PFC*. Laboratorio de Sistemas Integrados, ETSIT-UPM, Sep 2013.
- [116] Microchip Technology Inc., *MRF24XA Low-Power, 2.4 GHz ISM-Band IEEE 802.15.4<sup>TM</sup> RF Transceiver*. 2011.

# List of Acronyms

6LoWPAN.....	<i>IPv6 Low Power WPAN</i>
ACK.....	<i>Acknowledgment</i>
ADC.....	<i>Analog-to-Digital Converter</i>
API.....	<i>Application Programming Interface</i>
APL.....	<i>Academic Public License</i>
AWD.....	<i>Advanced Wireless Dynamics</i>
BER.....	<i>Bit-Error Rate</i>
BOM.....	<i>Bill of Materials</i>
CDC.....	<i>Communications Device Class</i>
CMOS.....	<i>Complementary Metal Oxide Semiconductor</i>
CN.....	<i>Cognitive Network</i>
CNGD.....	<i>Cognitive Next Generation Device</i>
CPU.....	<i>Central Processing Unit</i>
CR.....	<i>Cognitive Radio</i>
CRMODULE ...	<i>Cognitive Radio Module</i>
CRN.....	<i>Cognitive Radio Network</i>
CWSN.....	<i>Cognitive Wireless Sensor Network</i>
DAC.....	<i>Digital-to-Analog Converter</i>
DIE.....	<i>Departamento de Ingeniería Electrónica</i>
DMA.....	<i>Direct Memory Access</i>
DMIPS.....	<i>Dhrystone Million Instructions per Second</i>
DQI.....	<i>Data Quality Indicator</i>
DSA.....	<i>Dinamic Spectrum Access</i>
ED.....	<i>Energy Detect</i>

- EDA ..... *Electronic Design Automation*
- EEPROM..... *Electrically Erasable Programmable Read-Only Memory*
- EM..... *Electro-Magnetic*
- ETSIT..... *Escuela Técnica Superior de Ingenieros en Telecomunicación*
- FCC ..... *Federal Communications Commission*
- FCD ..... *First Cognitive Device*
- FFD ..... *Full Function Device*
- FHSS ..... *Frequency Hopping Spread Spectrum*
- FIFO ..... *First-in First-out*
- FP7 ..... *7th Framework Programme for Research and Technological Development*
- FPGA ..... *Field Programmable Gate Array*
- FSK..... *Frequency-Shift Keying*
- GPIO ..... *General Purpose Input-Output*
- GPRS..... *General Packet Radio Service*
- GSM..... *Global System for Mobile*
- HAL ..... *Hardware Abstraction Layer*
- I<sup>2</sup>C..... *Inter-Integrated Circuit*
- ICD ..... *In-Circuit Debugger*
- IDE ..... *Integrated development environment*
- IEEE..... *Institute of Electrical and Electronics Engineers*
- IP ..... *Internet Protocol*
- IRDA ..... *Infrared Data Association*
- ISM..... *Industrial, Scientific and Medical*
- ITU..... *International Telecommunication Union*
- KP ..... *Knowledge Plane*
- LED ..... *Light-Emitting Diode*
- Li-ION..... *Lithium-Ion*
- LoWPAN..... *Low-Rate Wireless Personal Area Network*
- LSI..... *Laboratorio de Sistemas Integrados*
- M2M..... *Machine-to-machine*

- MAC ..... *Medium Access Control*
- MCLR ..... *Master Clear*
- MCU ..... *Microcontroller*
- MII ..... *Media Independent Interface*
- MIMO ..... *Multiple-Input Multiple-Output*
- MIPS ..... *Microprocessor without Interlocked Pipeline Stages*
- MOS ..... *Metal-Oxide-Semiconductor*
- Ni-CD ..... *Nickel-Cadmium*
- Ni-MH ..... *Nickel-Metal Hydride*
- OFCOM ..... *Office of Communications*
- OSI ..... *Open Systems Interconnection*
- OTAP ..... *Over The Air Programming*
- OTG ..... *On-The-Go*
- PAN ..... *Personal Area Network*
- PCB ..... *Printed Circuit Board*
- PHY ..... *Physical*
- PIC ..... *Peripheral Interface Controller*
- PLL ..... *Phase-locked loop*
- QoS ..... *Quality of Service*
- QPSK ..... *Quadrature Phase-Shift Keying*
- QR ..... *Quick Response*
- RAM ..... *Random Access Memory*
- RF ..... *Radio Frequency*
- RI ..... *Radio Interface*
- RISC ..... *Reduced Instruction Set Computer*
- RMII ..... *Reduced Media Independent Interface*
- RSSI ..... *Received Signal Strength Indication*
- RTCC ..... *Real-Time Clock/Calendar*
- RX ..... *Reception*
- SDR ..... *Software-Defined Radio*

- SENDORA ... Sensor Network for Dynamic and Cognitive Radio Access
- SINR..... *Signal to Interference and Noise Ratio*
- SISO..... *Single-Input Single-Output*
- SMA..... *SubMiniature version A*
- SMD..... *Surface-Mount Devices*
- SN..... *Sensor Network*
- SNR..... *Signal to Noise Ratio*
- SOIC..... *Small Outline Integrated Circuit*
- SPI..... *Serial Peripheral Interface*
- SPICE..... *Simulation Program with Integrated Circuits Emphasis*
- SPST..... *Single Pole, Single Throw*
- SRAM..... *Static Random Access Memory*
- TKN..... *Telecommunication Networks Group*
- TTL..... *Transistor-Transistor Logic*
- TU..... *Technische Universität*
- TX..... *Transmission*
- UART..... *Universal Asynchronous Receiver-Transmitter*
- UPM..... *Universidad Politécnica de Madrid*
- USB..... *Universal Serial Bus*
- USRP..... *Universal Software Radio Peripheral*
- VCC..... *Virtual Control Channel*
- VGA..... *Video Graphics Array*
- WN..... *Wireless Network*
- WPAN..... *Wireless Personal Area Network*
- WRAN..... *Wireless Regional Area Network*
- WSN..... *Wireless Sensor Network*