International Conference on Automation, Control Engineering and Computer Science (ACECS)
Proceedings of Engineering and Technology – PET
Vol.20 pp.54 - 58

Implementation of a low cost SDR-based Spectrum Sensing Prototype using USRP and Raspberry Pi board

Soumaya El Barrak [12],Abdelouahid Lyhyaoui[2], Antonio Puliafito[1], Salvatore Serrano[1]
[1]*Engineering Department, University of Messina, Contrada di Dio - 98166 Messina, Italy*
[2]*LTI Laboratory, ENSA-Tangier, University of Abdelmalek Essaâdi, BP 1818 Tanger Principal, Tangier -Morocco*
elbarrak.soumaya@gmail.com, lyhyaoui@gmail.com
apuliafito@unime.it, sserrano@unime.it

Abstract— **Our paper presents a low cost experimental prototype for cognitive radio applications, which consists of a radio platform and a host cognitive engine, dedicated essentially to Wireless Sensor Networks (WSN) scenarios. It is carried out using 'USRP (Universal Software Radio Peripheral) and GNU-Radio' Software Defined Radio (SDR) platform, implemented on a single board computer, which is the Raspberry Pi 3 board. We have experienced the Spectrum Sensing mechanism, based upon Energy-Detection approach, due to its low computational complexity, and implementation availability within GNU-Radio environment. The result shows the sensing outputs data obtained by scanning wireless activities in the 2.4 GHz frequency band, necessary for the determination of the spectrum holes.**

Keywords- Cognitive Radio (CR), Software defined Radio (SDR), Spectrum Sensing, USRP, Gnuradio, Raspberry Pi, Energy Detector(ED)

I. INTRODUCTION

Among all wireless network resources, Radio Frequency Spectrum is revealed to be one of the most expensive and scarce resources. This is essentially due to the adoption of a static spectrum assignment policy, which gives to each specific licensed user, an exclusive use of a specific spectrum band. As a result, the unlicensed bands become overcrowded, especially, the ISM one, where the most ubiquitous communication protocols, like Wi-Fi and Bluetooth, operate. Thus, the quality of the communication will be affected due to the harmful interferences between different transceivers. Obviously, this will be followed by the increase of the energy consumption and the cost. This issue becomes more critical in a new city scale scenario, where the number of users and technologies increases, in order to concretize the newly coined IoT (Internet of Things) paradigm in creating next-generation cities, with a new conception of 5G network architecture as described in [1]. The key technology introduced to overcome the previous challenges related to the spectrum shortage problem is "Cognitive Radio Network". It aims at performing an opportunistic and dynamic use of the radio spectrum portions in the networks. The objective of our research activities is the implementation of a practical and soft Cognitive Radio (CR) Prototype, suitable to be used in a wireless sensor networks scenario, dedicated for smart cities applications.

Our prototype will be based upon a 'Software Defined radio' platform [2] run on a single-board computer as a cognitive engine where all the computational process will be held. In this paper, we are focusing on the Spectrum Sensing part, because a cognitive radio is a radio in which communication systems are aware of their environment, mainly, in terms of radio frequency (RF) spectrum occupancy at a specific time and location. For this reason, we started by the process of sensing, which is a primordial step to scan the radio frequency environment, in order to find spectrum holes. The sensing policy will employ the Energy Detection Algorithm (EDA) to determine the presence or the absence of a primary licensed user. This paper will be divided in six sections; the second one will present an overview about cognitive radio networks. The following one will describe the software and hardware platform of our prototype. In the fourth section, we will focus on the implementation of the energy detection algorithm on the software defined radio platform. The results obtained will be discussed in the fifth section. The last one will deliver the conclusion and future research activities.

II. COGNITIVE RADIO OVERVIEW

The idea of Cognitive radio networks was first proposed by Joseph Mitola in [3], who is considered as "the pioneer" of software radio and cognitive radio technologies. It aims at conceiving a radio device that can be programmed and configured to use the frequency spectrum bands intelligently and dynamically. It can automatically detect available channels in wireless spectrum, and change its transmission or reception parameters according to the information received from RF environment. The objective is using the spectrum more efficiently, by allowing unlicensed or 'secondary' users

ISSN 2356-5608

to share spectrum bands with the licensed or ‘primary’ users who own the property rights of exclusive usage of specific bands. This idea is encouraged by many statistical studies of the spectrum occupancy in some bands as described in [4] and [5], which show a low utilization of the scarce spectrum during the observation time at many locations. Thus, a considerable potential of secondary usage is attainable while the primary users are not transmitting.

To implement a CR prototype, we need a RF platform to interact with the RF environment along with a computational platform where all Digital Signal Processing (DSP) functions are held. Existing CR prototypes are mostly based on a cognitive engine (CE) controlling a SDR. Other research efforts were interested in building a low cost CR prototype based upon a frequency-agile and configurable platform but not imperatively a software-radio one [6]. Although, using this latter, we can not obtain the required degree of re-configurability as we can reach using SDR based prototype.

To achieve a dynamic spectrum access in a cognitive radio network, two policies must be taken into consideration [7]:

- Spectrum Sensing Policy: a cognitive user needs to gather the spectrum usage information via a spectrum sensing process prior to transmission, so as to decide whether a primary user is present or not.
- Spectrum Access Policy: this process aims at determining channel access related issues, according to the sensing results obtained, along with the wireless technology used for the communication.

In this paper, we are interested in building our CR experimental prototype, using the universal software radio platform USRP B200 from Ettus Research as a hardware package, along with GNU-Radio software package. The CE host machine will be a ‘Single Board Computer’, which is the Raspberry Pi 3 Model B. Concerning CR policies, we will focus on a spectrum sensing Energy-Detector based one, as first testing step of the experimental implementation.

III. DESCRIPTION OF THE PROTOTYPE COMPONENTS

Our idea was about proposing an efficient and low cost experimental prototype. For this reason, we opt to use a SDR platform, to guarantee the required degree of re-configurability, but it will be run on a modest single board computer instead of a desktop or a laptop. Thus, we will ensure a considerable balance between flexibility, cost, power consumption and size.

For the RF part of the prototype, we chose to use the USRP B200. It is a single channel radio transceiver board that covers continuously a wide frequency band between 60 MHz and 7GHz, including FM and TV broadcast, cellular, Wi-Fi and other signals. It has an open and reprogrammable Xilinx Spartan 6 FPGA along with 12 bits DAC/ADC converters. It can stream up to 56 MHz of real-time bandwidth over a USB 3.0 bus. The cognitive engine consists on the Raspberry Pi 3 board as a host machine, along with the software radio package ‘GNU-Radio’ which interacts with the USRP. This entity will hold all the computational tasks needed for the implementation of the energy detector. Ubuntu MATE 16.04[8] was adopted to be the operating system of our prototype, since it allows users to deploy a full operating environment for the RPi board with a large selection of available packages required for the system installation.

GNU-Radio [9] is an open source software development toolkit that operates the Radio from the user’s perspective. It provides signal processing blocks to implement software radios, with an easy to use graphical user interface. The applications are written and executed using python and C++ programming language.

USRP Hardware Driver (UHD)[10] is the base interface for all interactions between GNU-Radio and USRP board. UHD is an open-source, cross-platform driver that can run on Windows, Linux, and MacOS. It provides a common API, which is used by several software frameworks, including GNU- Radio. Figure 1 shows the architecture of the prototype, including the individual components and their functions.

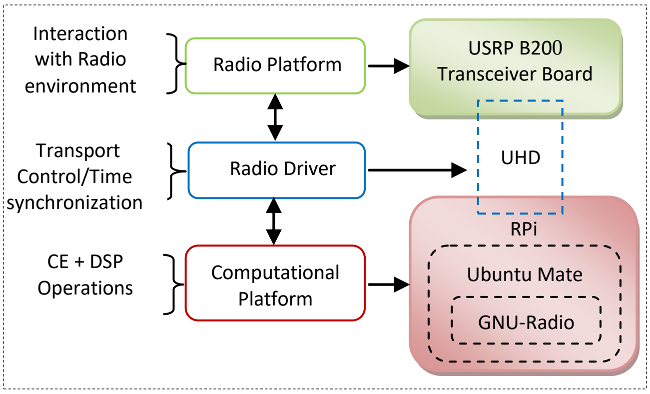

Fig 1: CR prototype architecture

IV. ENERGY-DETECTOR BASED SENSING

A. *Theory*

The Spectrum sensing process implemented is based on the energy detection algorithm in order to determine the presence or the absence of a primary user in the frequency band of interest. We opted for power measurement to minimize the computational complexities. The problem is formulated as a binary hypothesis according to the following model [11]:

$$x_i(t) = \begin{cases} n_i(t) & | H_0 \\ n_i(t) + s_i(t) & | H_1 \end{cases}$$

ISSN 2356-5608

- $x_i(t)$: The received signal at the CR user
- $s_i(t)$: The signal transmitted by the primary user (PU)
- $n_i(t)$: The additive white Gaussian noise (AWGN)
- H_0 : Hypothesis of the absence of the PU signal in the frequency band of interest.
- H_1 : Hypothesis of the presence of the PU signal in the frequency band of interest.

The identification of spectrum holes depends on spectrum scanning results received while applying the energy detector. In fact, to decide whether a PU is transmitting in a specific channel or not, an energy threshold must be chosen. If the energy received during the observation time, at a specific location is below the threshold, we can use the band for secondary usage. Otherwise, it will be considered occupied by a PU.

B. *Implementation Using SDR Prototype*

The project setup is implemented using GNU Radio installed on Ubuntu Mate 16.04 on the RPi board. The USRP B200 device is used to physically fetch the signal from real time environment. Communication between the radio board and the RPi board is achieved using USB cable. The signal passes through the following blocks in figure 2, before being processed in the host engine.

The project setup is implemented using GNU Radio installed on Ubuntu Mate 16.04 on the RPi board. The USRP B200 device is used to physically fetch the signal from real time environment. Communication between the radio board and the RPi board is achieved using USB cable. The signal passes through the following blocks in figure 2, before being processed in the host engine.

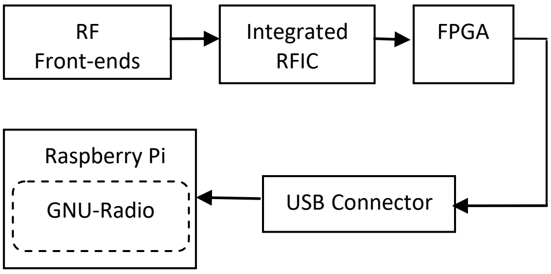

Fig. 2 : Hardware blocks for signal reception

The RF front-ends receive the RF signal and convert it to intermediate frequency (IF) one. Then, it is digitalized trough the ADC (Analog-to-Digital-Converter), and transferred to the FPGA, where digital down conversion functionalities are applied. The signal arrives to the host engine as a stream of discrete-time signal samples.

GNU-Radio SDR platform traits data received according to a predefined sequence of software blocks connected to each other, called flow graph. Figure 3 describes the processing mechanism of the energy detector.

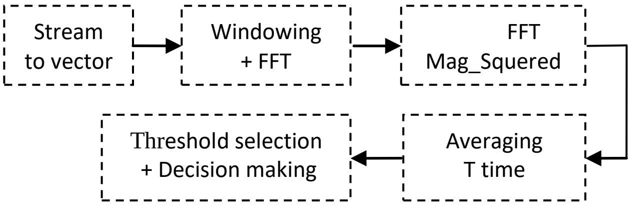

Fig. 3 : Energy-detector block-diagram

In each observation time, The 'Stream to Vector' block segments the stream of samples into V vectors. We consider $v_i(n)$ as the i^{th} vector with $i \in [\![1,V]\!]$, and $n \in [\![1,N]\!]$. N refers to the number of samples in a single vector. Then, Blackman-Harris window of size N is applied to each vector. Then, the Fast Fourier Transform (FFT) is evaluated for each windowed vector $v_{iw}(n)$, according to formula (1):

$$V_i(k) = \sum_{n=1}^{N} v_{iw}(n) e^{-j\frac{2\pi nk}{N}} \; ; \; k \in [\![1,N]\!] \qquad (1)$$

The magnitudes squared of the FFT block outputs are then calculated to obtain the PSD (Power Spectral Density) samples for each vector, according to formula (2):

$$S_i(k) = |V_i(k)|^2 \quad ; \; k \in [\![1,N]\!] \qquad (2)$$

Next step is averaging PSD samples over the total number V of vectors, as shown in the following formula (3):

$$S_{i_{average}}(k) = \frac{1}{V} \sum_{i=1}^{V} S_i(k) \; ; \; k \in [\![1,N]\!] \qquad (3)$$

Decision making based on the energy threshold selection is a critical phase in which two probabilities must be taken into consideration, in order to evaluate the performance of the previous sensing mechanism. The first one is the False-Alarm probability, which controls the cases when we receive erroneous detected energy levels. For instance, when the energy received is higher than the threshold even if the PU is absent. The second one is the detection probability, which controls the correct detection of energy levels that effectively correspond to the presence or absence of the PU. Performance evaluation process is not considered in this paper, because our goal is highlighting the conception and the design of the experimental prototype.

V. EXPERIMENTAL SETUP AND RESULTS

A. *Testbed Prototype*

Our SDR prototype is used to detect wireless activities in the 2.4 GHz band, based on the signal power received from the WiFi access points positioned in the seventh floor, bloc B, in the Engineering Department building of the University of Messina. Figure 4 shows the experimental test bed.

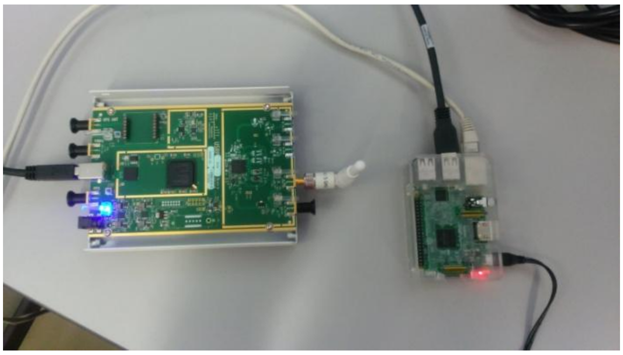

Fig. 4 : Experimental Testbed

The USRP is equipped with an antenna connected to the RX input port, which covers a frequency band from 2.4 to 5 GHz. We chose a sample rate equal to 4 MS/s and a tune delay of 0.25 s. This latter is a critical parameter which must be well chose because USRP B200 needs enough time to tune from one center frequency to another. In fact, when we command the USRP RF motherboard to change its center frequency, we have to wait until wanted ADC samples arrive to our FFT engine, which must belong to the wanted center frequency. This is done by dropping the incoming samples over a specified tuning delay time. This parameter will be used in the block *'**gr::blocks::bin_statistics_f**'* which controls scanning and records frequency domain statistics. The FFT-size is fixed to 1024 bins, and the noise floor is calculated according to formula (4), where *BW* is the channel bandwidth and '8' is the USRP's maximum receive noise figure. The obtained value is -128 (dBm).

$$NF_{dBm} = -174 + 10.\log(BW) + 8 \qquad (4)$$

B. *Sensing Results*

We applied the sensing mechanism in the band from 2.4 to 2.5 GHz, with the previous parameters. We received 15302 energy samples, each one corresponds to a frequency bin, collected by sweeping 99.2 MHz in the band of interest. The system divides the band to 19 sub-bands of 5.22 MHz. The difference between two consecutive frequency bins is fixed to 6250 Hz.

Our virtual WiFi analyzer has displayed eleven IEEE 802.11 networks occupying the channels from 1 to 13, along with the corresponding received power, as shown in figure (5).

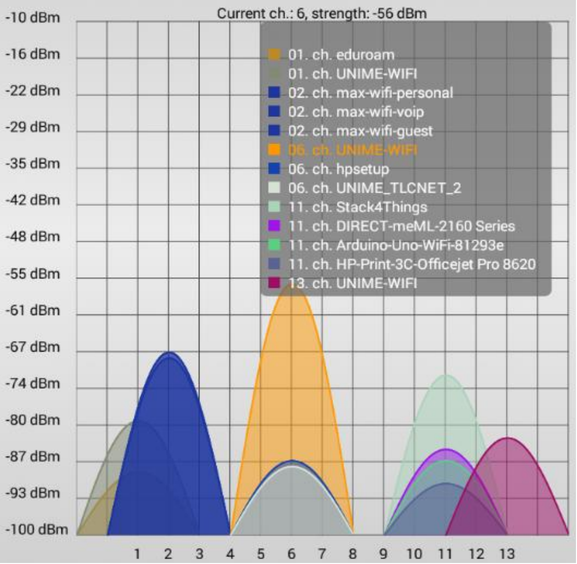

Fig. 5: IEEE 802.11 Channls occupancy

According to the WiFi analyzer, the networks detected occupy five essential channels, whose numbers are 1, 2, 6, 11 and 13. Each channel corresponds to a center frequency as detailed in the table (1):

Channel number	1	2	6	11	13
Center frequency (GHz)	2.412	2.417	2.437	2.462	2.472

Tab. 1 : Channel vs Center frequency

Figure (6) shows the obtained Power Spectrum Density (PSD) levels for each carrier frequency using the Energy-Detector.

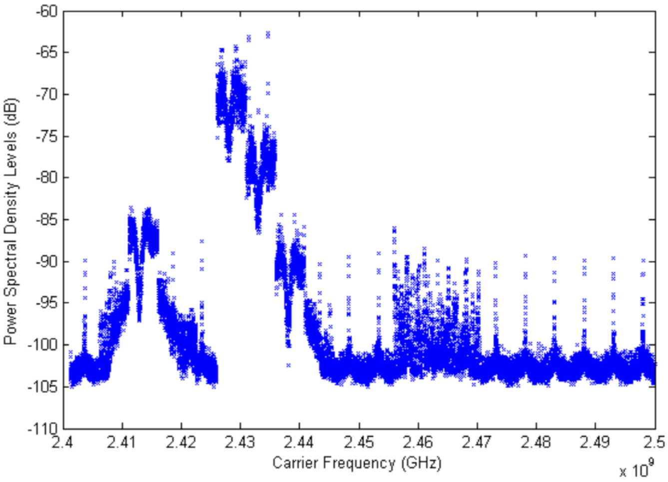

Fig. 6 : Spectrum Sensing in the 2.4 GHz band

ISSN 2356-5608

The power levels received are linearly-proportional to the instantaneous voltage seen at the antenna terminal, because samples coming from the USRP to GNU-Radio are rigorously calibrated to some standard amplitude by the hardware[12].

High power received signals are observed in 3 main zones. The first one corresponds to channel 1 and 2 where five networks are transmitting. The second zone corresponds to the channel 6 where three WiFi access points are transmitting, including the main network (UNIME-WIFI).Consequently, PSD levels received in this frequency band are higher than those received in the other bands. The third one refers to channel 11 and 13 where five access points are transmitting.

Thus, high power zones observed cover with compatibility, the frequency channels that know intensive wireless activities, whereas the spectrum holes correspond to the free bands where the PSD levels received are varying between -110 and -100 (dB).

VI. CONCLUSION

In this article we have proposed a simplified design of a CR prototype, based on SDR platform, which allows the radio software to run on a single board computer. It guarantees the balance between high configuration degree performed by using SDR platform, along with minimizing the energy consumption and the cost. The implementation of spectrum sensing mechanism in the 2.4 GHz frequency band, standing on Energy-Detection has been experienced. Decision making process and performance evaluation will be taken into consideration in our future research activities.

REFERENCES

[1] A. Gupta and R. K. Jha, “A Survey of 5G Network: Architecture and Emerging Technologies,” *IEEE Access*, vol. 3, pp. 1206–1232, 2015.

[2] G. Kolumbán, T. I. Krébesz, and F. C. M. Lau, “Theory and application of software defined electronics: Design concepts for the next generation of telecommunications and measurement systems,” *IEEE Circuits Syst. Mag.*, vol. 12, no. 2, pp. 8–34, 2012.

[3] J. Mitola and G. Q. Maguire, “Cognitive radio: making software radios more personal,” *IEEE Pers. Commun.*, vol. 6, no. 4, pp. 13–18, 1999 3.

[4] J. van de Beek, J. Riihijarvi, A. Achtzehn, and P. Mahonen, “TV White Space in Europe*,” IEEE Trans. Mob. Comput., vol. 11, no. 2, pp. 178–188, 2012.*

[5] K. Patil, K. Skouby, A. Chandra, and R. Prasad, “Spectrum occupancy statistics in the context of cognitive radio,” 2011 14th Int. Symp. Wirel. Pers. Multimed. Commun., pp. 1–5, 2011.

[6] A. R. Young and C. W. Bostian, “Simple and low-cost platforms for cognitive radio experiments [Application Notes*],” IEEE Microw. Mag., vol. 14, no. 1, pp. 146–157, 2013.*

[7] H. S. Kim, W. Ejaz, K. Al-Begain, A. S. K. Pathan, and N. U. Hasan, “Advances in cognitive radio sensor networks,” *Int. J. Distrib. Sens. Networks, vol. 2014, 2014.*

[8] https://ubuntu-mate.org/raspberry-pi/

[9] https://gnuradio.org/redmine/projects/gnuradio/wiki/Tutorials

[10] https://files.ettus.com/manual/

[11] T. Yucek and H. Arslam, “A Survey of Spectrum Sensing Algori-Athms for Congnitive Radio Applications*,” Proc. IEEE, vol. 97, no. 5, pp. 805–823, 2009.*

[12] https:// gnuradio.org/ redmine/ projects/gnuradio/wiki/FAQ##How-do-I-know-the-exact- voltagepower-of-my-received-input-signal

ISSN 2356-5608