REVISTA DIGITAL DE LAS TECNOLOGÍAS DE LA INFORMACIÓN Y LAS COMUNICACIONES
TELEM@TICA
Revista Telem@tica. Vol. 15. No. 1, enero-abril, 2016, p.69-79 ISSN 1729-3804

COGNITIVE RADIO PLATFORMS: A SURVEY.

Liz Martínez Marrero [1], David Alejandro Urquiza Villalonga [2], Eddy Hernández Inguanzo [3], Jorge Torres Gómez [4]

[1, 2, 3, 4] Departamento de Telecomunicaciones y Telemática, Facultad de Ingeniería Eléctrica, CUJAE, Cuba.

[1] liz.mm@electrica.cujae.edu.cu,, [2] barbicanci@infomed.sld.cu, [3] antoniohdez@infomed.sld.cu, [4] jorge.tg@electrica.cujae.edu.cu

ABSTRACT

The increasing demand of mobile users imposes a better utilization of the available spectrum. In this regard, Cognitive Radio (CR) technology represents a novel solution for avoiding spectrum scarcity. The implementation of the CR system follows two different approaches: Mitola Radio, which takes control of all parameters, and Cognitive Radio with just Spectrum Sensing (SS) techniques. This paper will be concerned with SS techniques. From a theoretical point of view, CR is addressed through several solutions in regard to novel SS methods. However, system and hardware development is progressing at a lower pace. Today, field-programmable gate arrays (FPGAs) and regular desktop computers are fast enough to handle complete baseband processing chains. There are several platforms, both open source and commercial, providing such solutions for supporting the development of novel SS methods. The aims of this paper is to give an overview of some of the available platforms and testbeds and their characteristics, and compare the features and performance measures of the different systems.

KEYWORDS: Cognitive Radio, reconfigurable platform, testbed, SDR.

RESUMEN

La creciente demanda de los usuarios de servicios móviles impone una mejor utilización del espectro radioeléctrico. En este sentido, la tecnología de Radio Cognitivo (CR) representa una novedosa solución para reutilizar el poco ancho de banda disponible. La implementación de los sistemas de CR sigue dos enfoques diferentes: Mitola Radio, que considera el control de todos los parámetros, y Radio Cognitivo solamente con Sensado de Espectro (SS). Este artículo se refiere a las técnicas de SS. Desde un punto de vista teórico, CR es abordado a través de varias soluciones en lo que respecta a novedosos métodos de SS. Sin embargo, el desarrollo de hardware y sistemas de desarrollo sucede a un ritmo inferior. Actualmente, los arreglos de compuertas lógicas programables (FPGA), así como los procesadores de propósito general han alcanzado el desarrollo suficiente para permitir el procesamiento en tiempo real de señales en banda base. Existen algunas plataformas, tanto de código abierto como comerciales, que ofrecen este tipo de soluciones sobre las que se apoya el desarrollo de nuevos métodos de SS. El objetivo de este artículo es brindar una revisión de las principales plataformas y sistemas de desarrollo de CR disponibles actualmente. Se exponen algunas de sus características y se comparan estas plataformas respecto a sus capacidades.

PALABRAS CLAVE: Radio Cognitivo, plataformas de desarrollo, SDR.

INTRODUCCIÓN

Currently, the increasing demand of mobile users imposes a better utilization of the available spectrum. In this regard, Cognitive Radio (CR) technology represents a novel solution for avoiding spectrum scarcity. CR, term first coined by Mitola [1], represents a communication system sufficiently intelligent for detecting user needs and employs radio resources accordingly. Taking into account the definition given by the UIT [2], CR must be implemented with three capabilities mainly: Cognitive, Re-configurable and Learning.

Two models are used for describing the CR operation, the simplified cognitive cycle [3] and the biologically inspired CR engine [4]. This paper addresses the simplified model, in which three cognitive task are performed [3]:

1. Radio-scene analysis, which encompasses the estimation of the interference temperature and the detection of spectrum holes.
2. Channel identification, which encompasses the channel-state information and prediction of channel capacity.
3. Transmit-power control and dynamic spectrum management.

On the other hand, the implementation of the CR system follows two different approaches [3]:

1. Full cognitive radio or Mitola Radio: Takes control of all parameters.
2. Cognitive Radio with just Spectrum Sensing (SS) techniques.

This paper will be concerned with SS techniques. In this manner, the improve of the spectrum utilization is affordable by sensing the different bands and detecting idle spectra, namely spectrum holes in time, frequency and space domains [5]. The general idea is to detect whether the spectrum is occupied by licensed Primary Users (PU), and then detect the PU activities for avoiding any interference.
From a theoretical point of view, CR is ad-dressed through several solutions in regard to novel SS methods. For instance, the Energy Detector, Cycloestationarity and Eigenvalue based sensing schemes has been suggested. However, system and hardware development is progressing at a lower pace [6]. This article summarizes the common used platforms and testbeds.

The platforms can be divided into three headings or categories:

3. Reconfigurable software/hardware systems: In these systems the majority of the radio functionality, such as modulation/coding/medium access is performed in software. The operations performed by the front end is intended to be minimal.
4. Composite Systems: This is comprised by a combination of pure software and hardware digital signal processing algorithms. This is attainable through the use of FPGA for example.
5. Standalone components: This components respond to the needs for spectrum sensing algorithms.

The rest of the paper is outlined as follows: Section 2 introduces the reconfigurable platforms. Section 3 illustrates the composite systems, including the standalone components. Finally Section 4 addresses the discussions and conclusions of this work.

RECONFIGURABLE PLATFORMS

The reconfigurable platforms are research oriented systems given by: GNURadio, OSSIE and Iris, which in turn are developed under GNU Licenses.

RF Front-Ends

Although these “Reconfigurable soft-ware/hardware systems” are designed to carry out the majority of signal processing in software they require a minimal hard-ware RF front-end. The most commonly used RF front-end, especially in the re-search world, is the Universal Software Radio Peripheral (USRP). The USRP is an inexpensive RF front-end and acquisition board with open design and freely available documentation and schematics. It is highly modular; a range of different RF daughterboards for selected frequency ranges may be connected.

Two types of USRP are available. USRP 1.0 contains four high-speed analog-digital converters (ADCs) supporting a maximum of 128 Msamples/s at a resolution of 14 bits with 83 dB spurious-free dynamic range, an Altera Cyclone FPGA for interpolation, decimation, and signal path routing, and USB 2.0 for the connection interface. USRP 2.0 replaces the Altera FPGA with a Xilinx Spartan 3-2000 FPGA, gigabit Ethernet, and an ADC capable of 400 Msamples/s with 16-bit resolution [6].

GNURadio

GNU Radio is the system with the most widespread usage [6]. This project begun in 2001 as a spin-off between the Massachusetts Institute of Technology (MIT) and the SpectrumWare project.

This useful platform provides wide variety of digital signal processing libraries. It is a free and open-source software licensed under GNU General Public License. It was designed with the aim of working in real time with software defined radios (SDR) offering great flexibility to the radio layout. All signal processing is done on the host PC. Hence, it has options to control SDR that allow to scan, syntonize, acquire, process and record radioelectric signals. Although it can be used without any hard-ware in a simulation environment for wire-less communications research [7] [8].

GNU Radio applications are designed by a graph flows. The nodes of such a graph are the processing blocks where the techniques to implement and the data flows are defined. Blocks can be written in C++ or in Python. Every block is realized to do one only task and corresponds to an XML file that describes its parameters, inputs, out-puts, and other attributes. GNU Radio offers several graphical sinks and graphical controls for creating wx-gui flow graphs. The variable control such as slider, text box and chooser block provides graphical widgets through which the user can change some parameter of the blocks in real time. Each of these graphical elements have a grid position parameter for precise positioning.

GNU Radio possess large numbers of blocks: filter, digital and analogic demodulator, FFT block, decoder, channel codes, synchronization elements, equalizers and others. Besides, it allows to develop novel signal processing codes and to include the obtained algorithm as a new block in your own design. This represents a very flexible environment.

The code is usually written in C++ language, in order to keep the platform as fast as it is. It can work with all kind of complex or real data format. GNU Radio has advantageous functionalities that guarantee a widely use in academic and commercial environments [9] [10] [11].

One of the main drawbacks of the platform is that the data processing latency is severe. The delays are mainly imposed by the flow graph block structure by which the software is designed, which leads to large buffers between each block. In addition, the host interface also imposes delays, especially for the USRP1, which cannot be easily overcome. Typical turnaround time from reception to transmission can reach several hundreds of microseconds. However, a stand-alone version of the USRP2 featuring a larger FPGA, where all the signal processing is done in the hard-ware, may solve the problem, although at an increased application development complexity [12].

OSSIE

The OSSIE Project is an initiative by Wire-less @ Virginia Tech to provide an Open-Source SCA (the Software Communications Architecture) Implementation::Embedded that allows hands-on learning at no cost. OSSIE was developed to facilitate education and research on SDR, cognitive radio, distributed wireless computing, and security. OSSIE is developed for Linux and specifically supports the Fedora and Ubuntu distributions. Cygwin or VMWare can be used to run OSSIE on Windows or other platforms [13].
Development and debugging tools were designed to facilitate the rapid prototyping of OSSIE components and waveforms in a visual environment. The complete set of tools is called the OSSIE Waveform Work-shop. It includes the OEF, OWD and ALF systems. The OEF (OSSIE Eclipse Feature) is a dedicated signal processing component and waveform development, this is an Eclipse plug-in and requires the installation of Java and Eclipse to run properly. The OWD (Waveform Developer) is a development tool that provides the same capabilities as OEF, however the application is standalone and only re-quires the installation of python to be executed. The ALF Graphical Debugging is a waveform blocks visualization and debugging environment for waveform and data visualization.
Additionally, when a waveform is been processed, the developer is able to monitor the throughput of components, plot data and manage input or output data to components in real time. ALF also gives to user the ability to run components that are not part of a host waveform in the framework. The Waveform Dashboard tool for execution and runtime configuration of wave-forms can be further customized to show or hide components and their properties on a per-waveform basis [13] [14].
OSSIE also provides a library of basic signal processing components to facilitate waveform development. The library includes: modulation and demodulation, interpolation and decimation, automatic gain control, carrier recovery, and symbol synchronization, among others. Furthermore, it uses a small subset of GNU Radio to communicate with and configure the USRP and USRP2 [13].

Iris

This is a dynamically reconfigurable soft-ware radio framework. This was developed by the University of Dublin, Trinity College. The system is comprised by a general purpose processor. The basic building block is comprised by a radio component written in C++, which implements one or more stages of a transceiver chain. The XML language is employed to specify the signal chain construction and characteristic. This plat-form provides dynamically processing algorithms for shaping and sculpting wave-forms in order to make best usage of avail-able white space. It also provides components for frequency rendezvous between two systems that are not known a priori. Besides, this system offers multiple sensing algorithms ranging from simple energy detection to mores sophisticated solutions through the use of filter banks and cicloestationarity [15].

COMPOSITE PLATFORMS

The main characteristic of composite plat-forms lies on reconfigurability. These systems contain all the required components that allow for immediate CR development. Example of such a systems are the BEE, WARP and KUAR platforms.

BEE

Named as the Berkeley Emulation Engine (BEE), this is a generic, multi-purpose, FPGA based, emulation platform for computationally intensive applications hard-ware developed at the University of California [16]. BEE2 consists of five Xilinx Vertex-II Pro VP70 FPGAs in a single compute module with 500 Giga-operations/s. These FPGAs can parallelize computationally intensive signal processing algorithms even for multiple radios. In addition to dedicated logic resources, each FPGA embeds a PowerPC 405 core for minimized latency and maximized data throughput between microprocessor and reconfigurable logic. The PowerPC on one of the FPGAs runs a modified version of Linux and a full IP protocol stack which allows to support protocol development and interfaces between other networked devices [6].

In order to interface this realtime processing engine with radios and other high throughput devices, multigigabit transceivers (MGTs) on each FPGA are used to form 10 Gb/s full-duplex links. So that, this sys-tem is well suited for real time applications with the interconnection of 18 independent radio links in an arbitrary network configuration [16].

BEE2 can be programmed using Matlab/Simulink from Mathworks coupled with the Xilinx system generator. The tool chain is augmented with BWRC developed automation tools for mapping high level block diagrams and state machine specifications to FPGA configurations. A set of parameterized library blocks have been developed for communications, control operators, memory interfaces and I/O modules [17, p. 2].

WARP

This system is also complete hardware and software design. The Wireless Open-Access Research Platform (WARP) was developed from Rice University, Houston, Texas in 2009. The platform architecture consists of four key components: custom hardware, platform support packages, open-access repository and research applications; all together providing a scalable, extensible and reconfigurable wireless testbed for students and faculty [12].

The hardware is comprised by a mother-board as an acquisition board, while daughterboards are used for data collecting. The processing is performed by a Xilinx Virtex-II FPGA module which provides significant processing resources to implement and test complicated physical (PHY) layers [12]. A second design flow, called WARPLab, can be used for system exploration, as in the case of cooperative communication algorithms. It allows rapid prototyping of physical layer algorithms over the air, by exposing WARP hardware to MATLAB [12].

KUAR

KUAR is given by the Kansas University Agile Radio. This is an experimental board in the range of 5.25 to 5.85 GHz, where the unlicensed national information infrastructure is placed [18]. It featured a Xilinx Virtex II Pro P30 FPGA with embedded PC for signal processing, four independent inter-faces between the FPGA and an embedded 1.4GHz general purpose processor. This approach allows split processing be-tween the embedded PC platform and FPGA. The platform uses an ADC with 105 Msamples/s and 14-bit resolution and sup-ports gigabit Ethernet and PCIexpress connections back to a host computer. This allows for all, or almost all processing, to be implemented on the platform, minimizing the host-interface communications requirements [19]. The platform was de-signed to be battery powered thus allowing for untethered operation. KUAR uses a modified form of the GNU Radio software framework to implement its signal processing features [6].

OpenAirInterface.org

OpenAirInterface (OAI) is an open experimentation and prototyping platform created by the Mobile Communications Department at EURECOM to enable innovation in the area of mobile/wireless networking and communications [20]. The platform uses all-IP wireless networking and implements the PHY and MAC layer in real-time on a combination of generic PC and FPGA-based equipment under open-source tool chains and software development environments.

The system is driven by two hardware components: CardBus MIMO I and Express MIMO. The first one is mostly software and PC-oriented, developed for targeting air interface experimentation support 2x2 MIMO communications. Express MIMO, showed in fig.1, is a generic baseband processing engine for high performance radio signal processing and aims at innovation in system-on-chip architectures for multimodal baseband subsystems. It makes use of two high density Xilinx Virtex 5 FPGAs: LX110T for providing a high speed (PCI-Express) interface with a PC based system and LX330, a signal processing engine, which can function as standalone. The prototype card contains four high-speed dual data converters (dual A/D and dual D/A) which can be connected to an external RF providing 8x8 MIMO input/output capacity [12].

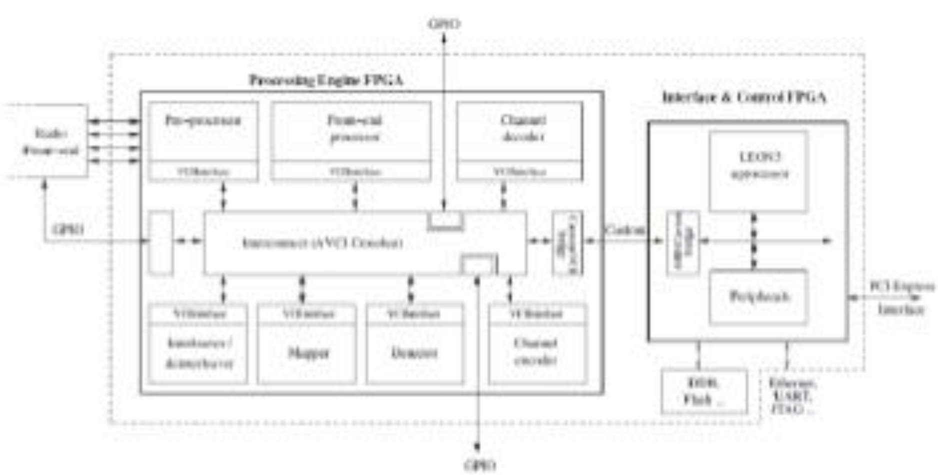

Figure1: Architecture of Express MIMO

WiNC2R

The architecture of the WiNC2R cognitive radio platform is developed by WINLAB at Rutgers University. This system addresses the workload characteristics of wireless communication protocols with programmable control mechanisms. Besides, this solution engage both hardware and software modules in a uniform manner in order to satisfy both functional and performance requirements.

The Virtual Flow Pipelining (VFP) architecture approach is designed to satisfy the workload characteristics and functional requirements of radio communication protocol processing applications. The main characteristics are the following:

a. System level control structure implemented in hardware for supporting the protocol processing flow requirements: function synchronization, scheduling, performance guaranties, sequencing, and communication;
b. The set of functional units is comprised of generic hardware modules and software programmable processors. Both are controlled by the system level Virtual Flow Pipeline controller;
c. c. Function switching are fast, on a per packet basis, regardless whether it is performed in hardware or software;
d. d. Hardware based central processing units scheduling is enable fast context switching and, hence, obviate the need for an embedded operating system [12].

The high-level architecture of the WiNC2R board is shown in fig. 2. The platform is featured by a baseband modem modules connected to the networking module. Each baseband module has associated an RF module.

The RF module uses double super-heterodyne driven by a pair of VCO/PLLs and switchable shaping filters for a wide tuning range of 30MHz to 6GHz with baseband of 0-500MHz. The module is half-duplex with planned full-duplex and skew/phase control reference clock extensions (for antenna arraying). In addition to the SPI control interface, all of the RF module features can be controlled directly by the (CR) host through the optional USB interface.

The baseband module provides two analog input and two analog output channels as interfaces to the RF module. Each pair of analog signals translates to a pair of I/Q data streams. The incoming analog signals are first low-pass filtered, sampled at a rate of 125MSPS, and then converted to a dual 14-bit data stream. This digitized version of the analog signal is further processed by the modem FPGA. The RF module frequency selection, Rx/Tx switching, and power settings are controlled through the provided SPI interface. These settings are configurable from the modem FPGA (Xilinx Virtex-4 SX series). A JTAG port is used for FPGA configuration downloads and debugging purposes. The networking module is implemented as a daughter board which can be plugged into the modem module. Packet and network protocol processing are performed using a Xilinx 4FX12 FPGA [21].

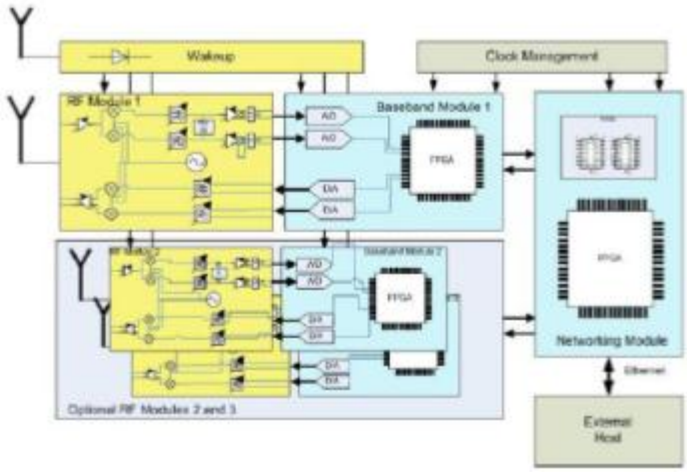

Figure 2: High level architecture of WiNC2R board [21]

Standalone components

The standalone components commonly implement SS algorithms in order to efficiently exploit the bandwidth resources and also to minimize interference. This is the case of the Rockwell Collins, IMEC and sensing devices from the Institute for Infocomm Research (I2R).

COBRA

IMEC introduces a cognitive baseband radio (COBRA) architecture targeting 4G requirements up to 1Gbit/s throughput and multiple asynchronous concurrent streams (for instance simultaneous digital broadcasting reception and high-speed internet access). This architecture can be customized to meet the requirements for many standards (WLAN (IEEE802.11n to .11ac), cellular (LTE to LTE-advanced), and broadcasting (DVB-T/H to DVB-T2)) and dedicated needs [21].

This platform largely consists of four types of cores:

a. DIFFS: A digital front end capable of sensing and synchronization;
b. ADRES: An energy-efficient high-performance reconfigurable baseband processor featuring multi-threading and wide SIMD (single instruction, multiple data) capabilities;
c. c. FlexFEC: A flexible forward error correction capable of doing different outer modem processing and;
d. d. ARM core for controlling the tasks on the platform.

All cores of this digital platform can be customized based on the requirements and the target standards that need to be supported. Besides, all the elements of the platform are programmable in C or high level assembly languages [14]. Moreover, COBRA features a novel ASIP (application-specific integrated processor)-based digital front-end enabling flexible filtering synchronization and spectrum sensing. This component also enables hierarchical platform activation, resulting in idle power in the range of 2mW in 65 low-power CMOS technology for the baseband platform [21].

NICT SDR platform

The Japanese National Institute of Information and Communications Technology (NICT) constructed a software-defined radio platform to trial next generation mobile networks. The platform had two embedded processors, four Xilinx Virtex2 FPGA, and RF modules that could support 1.9 to 2.4 and 5.0 to 5.3 GHz.

The SDR platform, as shown in fig. 3, featured an FPGA, CPU and RF boards. On the FPGA board, 4 FPGAs of 1152 pins are prepared. Two FPGAs process the physical layer signals for communication and broadcasting systems. This board is also connected with two analog-digital converters (ADCs), two digital-analog converters (DAC), and RF control units (RF cnt.). The RF control unit is an interface to manage the RF board, e.g. AGC, AFC, and channel selection. On the CPU board, two CPUs are prepared, a CPU controls system selection algorithm and TCP/IP stack and another controls MAC layer protocol. RF boards convert baseband signal to adequate frequency bands [22].

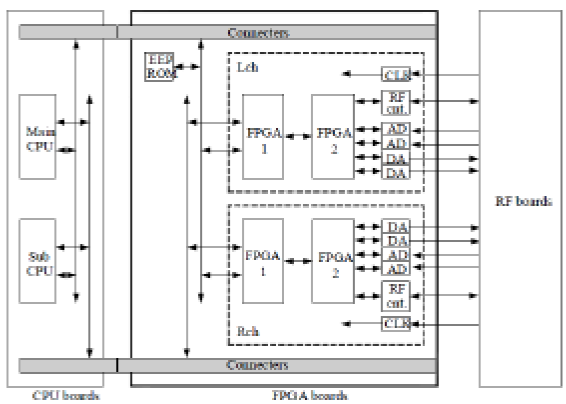

Figure 3: Configuration of NICT SDR platform

An objective of this platform was to explore selection algorithms to manage handover between existing standards. To this end, a number of commercial standards were implemented, for example, 802.11a/b/g, digital terrestrial broadcasting (Japanese format), wCDMA, and a general OFDM communication scheme [19].

CONCLUSIONES.

Cognitive radio offers an exciting and new way of thinking and researching in regard to wireless communication systems. It’s one of the promising techniques to meet the requirements for the future 5G network. Given that it exploits underutilized spectrum bands especially at microwave frequencies where there is a spectrum scarcity. For this reason, the platforms and testbeds have garnered attention as long as this is used to get closer to pragmatic implementations. Furthermore, a testbed can significantly speed up simulation and evaluation. By using existing architectures the development time is reduced.

In this work we have provided an overview of some different platform suitable for testbeds and demonstrators in the area of cognitive and cooperative communication. Three of them provide a platform for the development of reconfigurable cognitive radios networks. One of the main advantages of composite platforms is that they allow to evaluate new SS algorithms and to make over-the-air settings. Some of this platforms are openly available, GNU Radio, WARP, and OpenAirInterface, while others, such as WiNC2R and COBRA, are currently in internal use only.

REFERENCIAS.

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

[2] "Definitions of Software Defined Radio (SDR) and Cognitive Radio System (CRS)," ITU-R, SM.2152, Sep. 2009.

[3] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[4] Rieser, Christian James, "Biologically Inspired Cognitive Radio Engine Model Utilizing Distributed Genetic Algorithms for Secure and Robust Wireless Communications and Networking," Ph. D. Dissertation, 2004.

[5] A. Alizadeh, H. R. Bahrami, M. Maleki, and S. Sastry, "Spatial Sensing and Cognitive Radio Communication in the Presence of a -User Interference Primary Network," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 5, pp. 741–754, May 2015.

[6] X. Chen, H. H. Chen, and W. Meng, "Cooperative Communications for Cognitive Radio Networks #x2014; From Theory to Applications," *IEEE Commun. Surv. Tutor.*, vol. 16, no. 3, pp. 1180–1192, Third 2014.

[7] P. Pawelczak, K. Nolan, L. Doyle, S. W. Oh, and D. Cabric, "Cognitive radio: Ten years of experimentation and development," *IEEE Commun. Mag.*, vol. 49, no. 3, pp. 90–100, Mar. 2011.

[8] "WikiStart - GNU Radio - gnuradio.org." [Online]. Available: http://www.gnuradio.org/redmine/projects/gnuradio/wiki. [Accessed: 06-May-2016].

[9] "HowToUse - GNU Radio - gnuradio.org." [Online]. Available: http://gnuradio.org/redmine/projects/gnuradio/wiki/HowToUse#Using-the-included-tools-and-utility-programs. [Accessed: 06-May-2016].

[10] V. I. R. Abdala and J. S. Garcia, "Empowering Software Radio: IT++ AS A GNU Radio Out-Of-Tree Implementation," *IEEE Lat. Am. Trans.*, vol. 12, no. 2, pp. 269–276, Mar. 2014.

[11] S. H. Kabir, M. S. Omar, S. A. Raza, M. Hussain, and S. A. Hassan, "Demonstration and implementation of energy efficiency in cooperative networks," in *2015 International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2015, pp. 1341–1346.

[12] J. Mevada, J. Samuel, S. Bhadane, A. K. Gulati, and R. D. Koilpillai, "Design and implementation of a robust downlink communication system for nanosatellites," in *2015 International Conference on Space Science and Communication (IconSpace)*, 2015, pp. 164–169.

[13] O. Gustafsson, K. Amiri, D. Andersson, A. Blad, C. Bonnet, J. R. Cavallaro, J. Declerck, A. Dejonghe, P. Eliardsson, M. Glasse, A. Hayar, L. Hollevoet, C. Hunter, M. Joshi, F. Kaltenberger, R. Knopp, K. Le, Z. Miljanic, P. Murphy, F. Naessens, N. Nikaein, D. Nussbaum, R. Pacalet, P. Raghavan, A. Sabharwal, O. Sarode, P. Spasojevic, Y. Sun, H. M. Tullberg, T. V. Aa, L. V. der Perre, M. Wetterwald, and M. Wu, "Architectures for cognitive radio testbeds and demonstrators #x2014; An overview," in *2010 Proceedings of the Fifth International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, 2010, pp. 1–6.

[14] C. R. A. Gonzalez, C. B. Dietrich, and J. H. Reed, "Understanding the software communications architecture," *IEEE Commun. Mag.*, vol. 47, no. 9, pp. 50–57, Sep. 2009.

[15] Matt Carrick, Deepan Seeralan, Jason Snyder, Haris Volos, Drew Cormier, Christopher Covington, Carl B. Dietrich, Joseph Gaeddert, Benjamin Hilburn, C. Ian Phelps, and Shereef Sayed, "OSSIE 0.8.2 Installation and User Guide." Apr-2011.

[16] P. D. Sutton, J. Lotze, H. Lahlou, S. A. Fahmy, K. E. Nolan, B. Ozgul, T. W. Rondeau, J. Noguera, and L. E. Doyle, "Iris: an architecture for cognitive radio networking testbeds," *IEEE Commun. Mag.*, vol. 48, no. 9, pp. 114–122, Sep. 2010.

[17] "BEE2 - Berkeley Emulation Engine 2." [Online]. Available: http://bee2.eecs.berkeley.edu/. [Accessed: 17-Jun-2016].

[18] S. M. Mishra, D. Cabric, C. Chang, D. Willkomm, B. van Schewick, A. Wolisz, and R. W. Brodersen, "A real time cognitive radio testbed for physical and link layer experiments," in *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005.*, 2005, pp. 562–567.

[19] R. Farrell, M. Sanchez, G. Corley, R. Farrell, M. Sanchez, and G. Corley, "Software-Defined Radio Demonstrators: An Example and Future Trends, Software-Defined Radio Demonstrators: An Example and Future Trends," *Int. J. Digit. Multimed. Broadcast. Int. J. Digit. Multimed. Broadcast.*, vol. 2009, 2009, p. e547650, Mar. 2009.

[20] G. J. Minden, J. B. Evans, L. Searl, D. DePardo, V. R. Petty, R. Rajbanshi, T. Newman, Q. Chen, F. Weidling, J. Guffey, D. Datla, B. Barker, M. Peck, B. Cordill, A. M. Wyglinski, and A. Agah, "KUAR: A Flexible Software-Defined Radio Development Platform," in *2007 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, 2007, pp. 428–439.

[21] Z. Miljanic, I. Seskar, K. Le, and D. Raychaudhuri, "The WINLAB Network Centric Cognitive Radio Hardware Platform - WiNC2R," in *2007 2nd International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, 2007, pp. 155–160.

[22] H. Harada, "Software defined radio prototype toward cognitive radio communication systems," in *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005.*, 2005, pp. 539–547.