



## Evolution from SDR to Cognitive Radio

### KEYWORDS

Software Defined Radio, Cognitive radio, radio frequency

**Shilpa Jain**

Electronics and Communication dept.  
Inderprastha Engineering College  
Ghaziabad, U.P, India

**Nidhi Taneja**

Electronics & Communication Engineering dept.  
Delhi Technological University  
Delhi, India

### ABSTRACT

*Software Defined Radio (SDR) is a flexible radio architecture which can be configured to adapt various waveforms, frequency bands, bandwidths, modes of operations and wireless standards simply by altering the physical layer behavior through changes in its software. This paper presents a detailed survey of the existing hardware and software platform for SDRs. However, an SDR can switch functions and operations only on demand; it is not capable of reconfiguring itself into the most effective form without its user even knowing it. Therefore, Cognitive radio (CR) came into existence which extends the software radio with radio-domain model-based reasoning and would be trainable in a broad sense, instead of just programmable. In this paper a survey of spectrum sensing methodologies for cognitive radio is presented. These cognitive technologies may be considered as an application on top of a basic SDR platform.*

### I. INTRODUCTION

In a remarkably visionary article published in 1993 [1], Joseph Mitola III envisioned a very different kind of radio: A digital radio that could be reconfigured in fundamental ways just by changing the software code running on it. He dubbed this as software-defined radio. A few years later Mitola's vision turned into reality. In the mid-1990s military radio systems came into existence in which software controlled most of the signal processing digitally, enabling one set of hardware to work on many different frequencies and communication protocols. SPEAKeasy I and SPEAKeasy II radios [2], which allowed units from different branches of armed forces of U.S. military to communicate, were the first known examples of this type of radio.

In the late 1990s SDR started to spread from the military domain to the commercial sector. Cellular networks were considered as the most obvious and potentially most lucrative market that SDR could penetrate. The benefits it could bring to this industry included a general-purpose and therefore more economic hardware platform, easier bug fixes through software upgrades, and increased functionality and interoperability through the ability to support multiple standards.

The reconfigurability offered by SDR technology can be achieved only on demand; it is not capable of reconfiguring itself into the most effective form without its user even knowing it. In his licentiate thesis Mitola introduced Cognitive radio (CR) [3],[4] as "a really smart radio that would be self-aware, RF-aware, user-aware, and that would include language technology and machine vision along with a lot of high-fidelity knowledge of the radio environment". Cognitive radio clearly goes hand in hand with SDR; together, they can achieve functionality considered impossible only a decade ago. These cognitive technologies may be considered as an application on top of a basic SDR platform. Consequently, before continuing any further with respect to CR, we first provide an overview of SDR technology in Section II. This is followed by a detailed survey of various hardware and software platforms for SDR in Section III. An introduction of Cognitive radio is presented in section IV. Finally, various Spectrum sensing functionalities especially

of non-cooperative types are discussed in Section V.

### II. SOFTWARE DEFINED RADIO

A Software Defined Radio (SDR) the baseband operation characteristics of the radio, such as coding, modulation/demodulation, error correction coding, comp type and frequency band, can be changed at will, simply by loading new software. The multiple radio devices using different modulations can be replaced by a single radio device that can perform the same task. However, there are a number of challenges in the transition from hardware radio to software defined radio. First, transition from hardware to software processing results in a substantial increase in computation, which in turn results in increased power consumption. This reduces battery life and is one of the key reasons why software-defined radios have not been deployed yet in end-user devices, but rather in base stations and access points, which can take advantage of external power resources. Second, the AD/DA conversion should be moved as close as possible to the antenna so that all signal processing can be done digitally and an ideal SDR can be realized. Taken together, this means that high-bandwidth, high-frequency RF transmissions require very high sampling rates. Indeed, it is only recently that sufficiently fast DSPs and wideband AD/DA chipsets have become available at affordable cost to make it feasible to contemplate AD conversions of the IF rather than the baseband signal. SDR is currently used to build radios that support multiple interface technologies such as GSM, CDMA and WiFi with a single modem by reconfiguring it in software [5].

#### A. Real SDR Model

The dominant implementation architecture used for RF Front-Ends (FEs) is the super-heterodyne architecture [6],[7]. The figure 1 shows the model of real SDR system. The antenna receives the analog radio signal. An intermediate step before conversion is needed in the receiver. This conversion to an intermediate frequency is required since SDRs must deal with radio frequency signals. This step transforms the received high-frequency signal into a so called Intermediate Frequency (IF) by a tuner. Following this the IF is filtered and digitized. The filtering is done

to prevent aliasing frequency signals into the band of frequencies that are being digitized. The stream is received and processed in a combination of software and hardware. These hardware and software process the waveform. An output waveform is sent as a digital signal to be converted by DAC into an analog signal. A similar transformation can be made to shift the IF back for transmission. The analog signal is generally amplified and transmitted into air by a radio antenna.

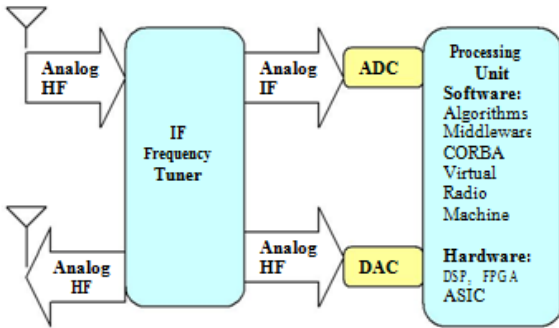


Fig. 1. Real SDR Model

**B. SDR Receiver**

The figure 2 shows a block diagram of a SDR receiver. The RF tuner converts analog RF to analog IF signals. The A/D converter that follows digitizes the IF signal thereby converting it into digital samples. These samples are fed to the next stage which is the digital down converter (DDC) shown within the dotted lines. The DDC is typically a single monolithic chip or FPGA IP, and it is a key part of the SDR system.

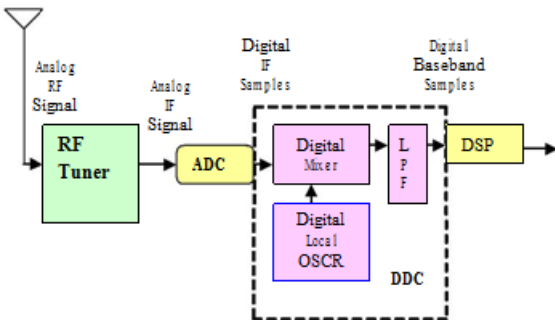


Fig. 2. SDR Receiver

The digital mixer and local oscillator translate the digital IF samples down to baseband. The FIR low pass filter limits the signal bandwidth and acts as a decimating low pass filter. The digital baseband samples are then fed to a block labeled DSP, which performs tasks such as demodulation, decoding and other processing tasks.

**C. SDR Transmitter**

The input to the transmitter side of an SDR system is a digital baseband signal, typically generated by a DSP stage after processing, is shown in figure 3. The digital hardware block in the dotted lines is a DUC (digital up-converter) that translates the baseband signal to the IF. The D/A converter that follows convert the digital IF samples into the analog IF signal. Next, the RF upconverter converts the analog IF signal to RF frequencies. Finally, the power amplifier boosts signal energy to the antenna .

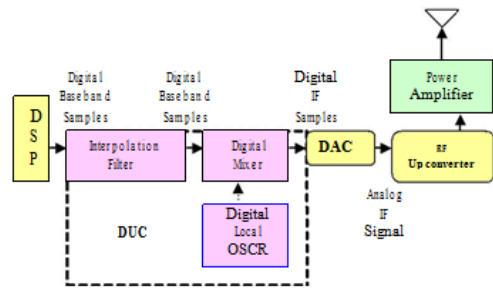


Fig. 3. SDR Transmitter

**D. SDR Modules and Hardware**

The main hardware alternatives that can be used to implement a SDR are:

- ASICs (Application-Specific Integrated Circuits).
- FPGAs (Field-Programmable Gate Arrays).
- DSPs (Digital Signal Processors).
- GPPs (General-Purpose Processors).

Table.1 shows the comparison between DSP, ASICs, GPPs and FPGAs. DSPs are microprocessors with architecture, instructions and features suited specifically for signal processing applications. DSP and GPPs are essentially serial in operation. The main strengths of DSPs and GPPs are their flexibility and easy configurability. Field Programming Gate Arrays (FPGA) contains DSP blocks that can be re-configured to work as parallel multiplier/adder or MAC. FPGA are extremely flexible and fast as they provide high computing power due to quasi-parallel processing nature. The ASICs are non-reprogrammable that contradicts the principle of SDR, but still used as a part for special characteristics.

TABLE I. COMPARISON OF ASIC, DSP, FPGA & GPP

| Comparison Parameters | High-Speed DSPs | Multiple ASICs | GPP      | FPGAs         |
|-----------------------|-----------------|----------------|----------|---------------|
| Power Consumption     | Very High       | Very Low       | Moderate | Low           |
| Size                  | Modest          | Large          | Moderate | Low           |
| Cost                  | Moderate / High | High           | Moderate | Moderate /Low |
| Field Upgradable      | High            | None           | Some     | High          |

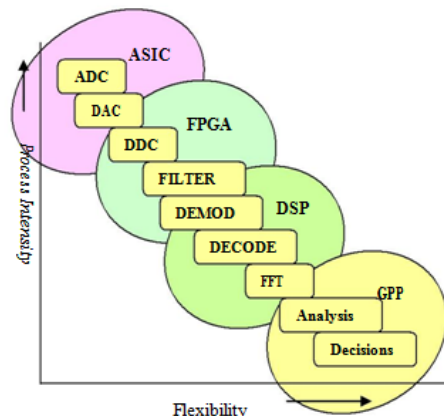


Fig. 4. ASIC, FPGA, GPP and DSP Platform for different SDR Modules

The graph of Processing Intensity vs. Flexibility as shown in figure 4 highlights some of the popular signal processing tasks associated with SDR system [8]. Processing intensity is the degree of highly repetitive operations. The upper left area indicates dedicated functions like ADC & DDC that requires specialized hardware structures like ASIC and FPGA, to complete the operations in real time. Flexibility defines how easily the functionality can be changed or customized for any specific applications. The lower right area shows functions like analysis and decision making which are highly variable & often subjective, therefore requires programmable GPP for this purpose. Intermediate area shows modules like filter, Modulator/Demodulator, Encoder/Decode which needs programmability as well as fast computation and can be implemented through DSPs or GPPs. Software Defined Radio platforms.

There are various hardware platforms and the software architectures that are used for defining the software radios [8],[9]. This section presents a survey of the current SDR hardware platforms followed by the software architectures.

**E. SDR Hardware Platforms**

Table II shows a detailed survey of existing SDR hardware platforms (Front end) and their performance.

**1) Universal Software Radio Peripheral 2 (USRP2)**

It is a product of Matt Ettus (Ettus Research LLC) [10]. The USRP2 platform is a second generation of Universal Software Radio Peripheral [10]. USRPs are commonly used with the GNU Radio software suite to create complex software-defined radio systems.

**2) Rice Wireless Open-Access Research Platform (WARP)**

The wireless open-access research platform of Rice University is a scalable and extensible programmable platform, built for prototyping advanced wireless networks [11]. It has programmability of both physical and network layer protocols on a single platform.

**3) Berkeley Emulation Engine 3 (BEE3)**

BEE3 is new generation of Berkeley Emulation Engine-2 [12].It is jointly developed by Microsoft Research, UC Berkeley and BEE cube Inc. It is useful for most computationally intensive real-time applications, high-speed multiple FPGA, real-world prototyping and development platform.

**4) Kansas University Agile Radio (KUAR)**

The KUAR hardware [13] has been promoted through the defense advanced research projects agency (DARPA) as next generation (XG) program. The complete system was developed in Simulink and implemented in Xilinx VHDL by generating the VHDL code from Simulink model(s) using a Modelsim of Mentor Graphics.

**5) Small Form Factor Software Defined Radio (SSF-SDR)**

The Xilinx Inc. in collaboration with Lyrtech and Texas Instruments incorporated a SFF-SDR development platform for developing the handheld and mobile radios [14].

**6) Intelligent Transport System (ITS)**

National Institute of Information and Communications Technology (NICT) of Japan, developed a software-defined radio platform so-called NISTITS. It is specially designed for mobile communication, wireless LAN and digital terrestrial TV [15].

**F. SDR Software Platforms**

**1) GNU Radio**

It is an open source software development toolkit that provides the signal processing runtime and processing blocks to implement software radios using readily-available, low-cost external RF hardware and commodity processors [10]. The radio applications are written in Python, while the performance critical signal processing components are implemented in C++. GNU Radio Companion (GRC) is a graphical tool for creating signal flow graphs and generating flow-graph source code. Thus, the developer is able to implement real-time, high throughput radio systems in a simple-to-use, rapid-application development environment

|                              | TABLE II.                                      |  | A SURVEY OF SDR HARDWARE PLATFORMS [13-21] |   |  |                                    |
|------------------------------|--|--|--|---|--|------------------------------------|
|                              | USRP2  | KUAR                                   | WRAP v2.2                                  | SSF-SDR                                   | BEE3   | NICT-ITS                           |
| Architecture (GPP,DSP, FPGA) | FPGA (Xilinx Spartan III) GPP (AeMB Processor) | FPGA (Xilinx-Virtex11) GPP (Pentium M) | FPGA (XilinxVirtex-4) GPP(PowerPC)         | FPGA (Xilinx Virtex- 4)                   | 4FPGAs (Xilinx Virtex-5) Quad-Core OpenSPARC | FPGA (Xilinx Virtex-4) GPP(ARM 11) |
| RF Bandwidth                 | 100 MHz  | 30 MHz                                 | 30 MHz                                     | 22 MHz                                    | 100 MHz                                      | 5 MHz                              |
| RF Range                     | 24 GHz & 5 GHz (multi M/G Hz)                  | 24 GHz & 5 GHz ISM/UNII                | 24 GHz & 5 GHz ISM/UNII                    | 0.2-1.0, 1.6-2.2 GHz, 2.5/3.5 GHz (WiMAX) | Ultra-wide-band (multi-GHz)                  | 2 GHz ,5 GHz and VUHF              |
| RF Channels                  | 2  | 16                                     | 4  | 2   | 16   | 5                                  |
| Connectivity                 | Gig Ethernet (National Semiconductor)          | USB 2.0, Gig Ethernet                  | USB 2.0, Gig Ethernet(Marvel)              | RS232/ USB 2.0, Gig Ethernet              | RS232/USB Gig Ethernet Broadcom              | RS232/USB Ethernet                 |
| ADCs                         | 14-bit ,100 MS/s (LTC2284)                     | 14-bit ,105 MS/s (LTC2284)             | 14-bit,65MS/s (AD9248)                     | 14-bit, 125MS/s (ADS5500)                 | 8-bit,3GS/s (BEE3-ADC-D3G)                   | 12-bit, 170MS/s (ADCs N/C)         |
| DACs                         | 16-bit,400MS/s (AD9777)                        | 16bit, 100MS/s (AD9777)                | 16-bit,160MS/s (AD9777)                    | 16-bit,500MS/s (TI DAC5687)               | 12-bit, 2GS/s (BEE3-DAC-D2G)                 | 12-bit, 500MS/s (DAC N/K)          |
| Power                        | 6 volt   | 12 volt                                | 12 volt                                    | 12 volt                                   | 12 volt                                      | 3.3 volt                           |
| Price                        | \$ 1400.00                                     | \$4,000.00                             | \$6,500.00                                 | 99,900.00                                 | \$20,000.00                                  | N/A                                |

2) Open-Source SCA Implementation Embedded (OSSIE) It is a Virginia Tech's open source, the core framework is based on the JTRS software communications architecture (SCA); the CORBA based communication model for SDR [16]. The OSSIE is an object-oriented SCA operating environment, where signal processing components are written in C++. The operating environment, often referred to as the core framework, implements the management, configuration, and control of the radio system. Every OSSIE's component is considered having two parts: one part realizing the signal processing and another managing the SCA infrastructure. The OSSIE waveforms are described in an XML that is used to describe component properties and interconnections between components in a waveform.

### 3) Wireless Open-Access Research Platform for Network

It is an SDR framework that is built around client server architecture in Python [17]. The WARPnet uses PCAP (packet capture) API to communicate with the WARP board directly. To allow the Python-based client/server to access PCAP, the Pcap module is required. With WARPLab, one can interact with WARP nodes directly from the MATLAB workspace and signals generated in MATLAB can be transmitted in real-time over-the-air using WARP nodes.

### 4) Cognitive Radio Open Source System (CROSS)

It is open source cognitive radio architecture [18]. It consists of five core components categories (modules); cognitive radio shell (CRS), cognitive engine (CE), policy engine (PE), service management layer (SML), and software-defined radio host platform. The CROSS is a modular cognitive radio system framework that uses socket connections for inter-component communication. The cognitive radio shell library and API are implemented in C++, the other modules can be implemented in any language that supports a TCP/IP socket interface.

## III. COGNITIVE RADIO

The need for higher data rates is increasing as a result of the transition from voice-only communications to multimedia type applications; therefore it becomes obvious that the current static frequency allocation schemes cannot accommodate the requirements of an increasing number of higher data rate devices. As a result, Cognitive radio [19] arises to be a tempting solution to the spectral congestion problem by introducing opportunistic usage of the frequency bands that are not heavily occupied by licensed users. In this paper, we use the definition adopted by Federal Communications Commission (FCC) [20]: "Cognitive radio: A radio or system that senses its operational electromagnetic environment and can dynamically and autonomously adjust its radio operating parameters to modify system operation, such as to mitigate interference, maximize throughput, facilitate interoperability and access secondary markets." [20]. Hence, one major feature of cognitive radio is its dynamic Spectrum Management principle which solves the issue of spectrum underutilization in wireless communication in a better way.

In cognitive radio terminology, primary users (PU) (also known as licensed device) can be defined as the users who have higher priority or legacy rights on the usage of a specific part of the spectrum and the secondary users (also called cognitive radio users or unlicensed devices), which have lower priority, exploit this spectrum in such a way that they do not

cause interference to PUs and they vacate the band once the PU is detected. Therefore, secondary users need to possess cognitive radio capabilities, such as sensing the spectrum reliably to check whether it is being used by a PU and to change the radio parameters to exploit the unused part of the spectrum called spectrum holes or white spaces, as shown in figure 5. Such kind of reconfiguring to adapt various waveforms, frequency bands, wireless standards, bandwidths, and modes of operations by altering the physical layer behavior can be easily achieved by SDRs, through changes in its software. Therefore SDR provides an ideal platform for the realization of Cognitive radio. Hence, Cognitive radio extends the software radio with radio-domain model-based reasoning.

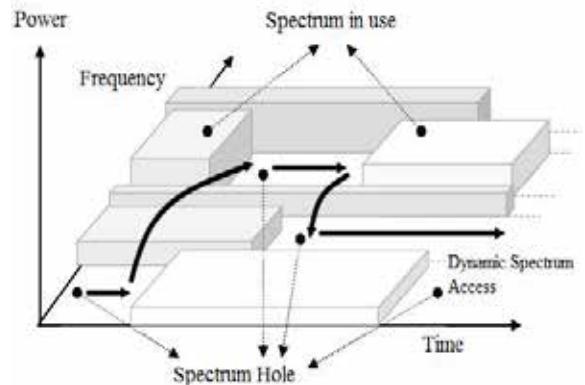


Fig. 5. Spectrum hole concept

Cognitive radio has four major functions [5]. They are Spectrum Sensing, Spectrum decision, Spectrum Sharing and Spectrum Mobility. Spectrum Sensing is to identify the presence of licensed users and unused frequency bands i.e. white spaces in those licensed bands. Spectrum decision is to identify how long the secondary users can use those white spaces. Spectrum Sharing is the fair sharing of white spaces (spectrum hole) among the secondary users (CRs). Spectrum Mobility is to maintain unbroken communication during the transition to better spectrum.

## IV. SPECTRUM SENSING FUNCTIONALITIES FOR COGNITIVE RADIO

The major challenge of the cognitive radio is that the secondary user needs to detect the presence of primary user and to quickly quit the frequency band if the corresponding primary radio emerges in order to avoid interference to primary users. Spectrum sensing can be classified [21] as shown in figure 6 as:

### A. Spectrum Sensing for Spectrum opportunities

#### 1) Primary transmitter detection:

Based on the received signal at CR users the detection of primary users is performed. This approach includes Matched filter (MF) detection, Energy detection (ED), Covariance detection, Waveform detection and Cyclostationary detection.

#### 2) Cooperative and collaborative detection:

The primary signals for spectrum opportunities are detected reliably by interacting or cooperating with other users, and the method can be implemented as either centralized access to spectrum coordinated by a spectrum server or distributed approach implied by the spectrum load smoothing algorithm or external detection.

**B. Spectrum Sensing for Interference Detection**

**1) Interference temperature detection:**

In this approach, CR system works as in the ultra wide band (UWB) technology where the secondary users coexist with primary users and are allowed to transmit with low power and are restricted by the interference temperature level so as not to cause harmful interference to primary users.

**2) Primary receiver detection:**

In this method, the interference and/or spectrum opportunities are detected based on primary receiver's local oscillator leakage power.

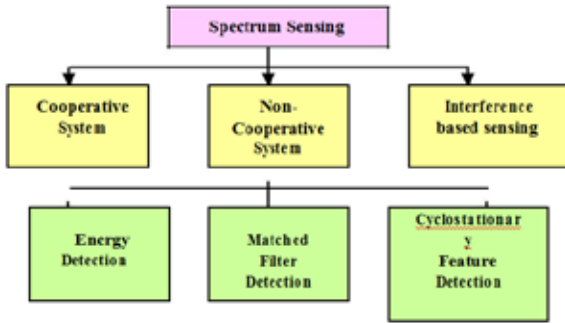


Fig. 6. Classification of Spectrum Sensing techniques

**A. Primary Transmitter Detection:**

In this we are going to discuss about few primary transmitter detection techniques. They are:

**1) Energy Detection(ED):**

In this technique there is no need of prior knowledge of primary signal energy [22].

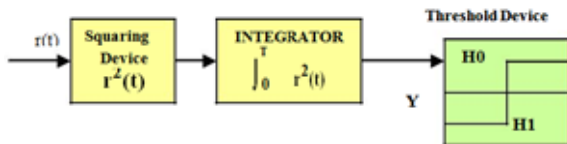


Fig. 7. Block Diagram of Energy Detection

Where  $H_0$  = Absence of User.  
 $H_1$  = Presence of User.

As depicted in figure 7, the measured signal  $r(t)$  is first squared and then integrated over the observation interval  $T$ . The output from the integrator block is then compared to a predefined threshold level  $\lambda_e$ . By this comparison we discover the presence or absence of the primary user. The threshold value can be fixed or variable based on the channel conditions. Here we consider two hypotheses:

$H_0: y(k) = n(k)$

$H_1: y(k) = h * s(k) + n(k)$

where  $y(k)$  is the sample to be analyzed at each instant  $k$  and  $n(k)$  is the noise of variance  $\sigma^2$ . Let  $y(k)$  be a sequence of received samples  $k \in \{1, 2, \dots, N\}$  at the signal detector, then a decision rule can be summarized with two probabilities:

**a) Probability of detection PD:**

It is the probability of detecting a signal on the considered frequency when it is truly present. It can be written as

$PD = Pr (M > \lambda E|H_1)$

**b) Probability of false alarm PF :**

It is the probability that the test incorrectly decides that the considered frequency is occupied when it actually is not, and it can be formulated as

$PF = Pr (M > \lambda E|H_0)$

where  $Pr$  is the probability of reception and the decision metric for the energy detector can be written as

$M = \hat{a}_n^N y(n)^2$

PF should be kept as small as possible to prevent underutilization of transmission opportunities. The decision threshold  $\lambda E$  can be selected for finding an optimum balance between PD and PF .

**2) Matched Filter:**

A Matched Filter (MF) is a linear filter designed to maximize the output signal to noise ratio for a given input signal [23]. When secondary user has a priori knowledge of primary user signal, matched filter detection is applied. The block diagram for the Matched filter detection technique is shown in the figure 8.

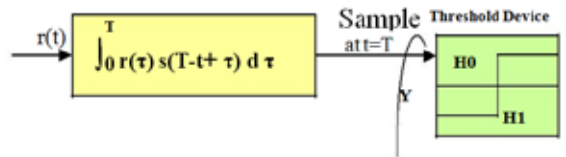


Fig. 8. Block Diagram of Matched Filter detector

Matched filter operation is equivalent to correlation in which the unknown signal is convolved with the filter whose impulse response is the mirror and time shifted version of a reference signal. The operation of matched filter detection is expressed as:

$Y[n] = \sum h[n-k] x[k]$

Where „ $x$ “ is the unknown signal (vector) and is convolved with the „ $h$ “ which is the impulse response of the matched filter that is matched to the reference signal for maximizing the SNR. Detection by using matched filter is useful only in cases where the information about primary users signaling features such as bandwidth, operating frequency, modulation type and order, pulse shaping, and frame format are known to the cognitive users in advance.

**3) Cyclostationary detection:**

In feature detection the presence of PU signals is determined by extracting their specific features such as pilot signals, symbol rate, cyclic prefixes, spreading codes, or modulation types from its local observation. These features can be detected by analyzing a spectral correlation function since they introduce built-in periodicity in the modulated signals, which is shown in figure 9. This is also called Feature detection [24]. Here, the spectrum correlation of the received signal  $r(t)$  is averaged over the interval  $T$ , and compared with the test statistic to determine the presence of PU signals, similar to energy detection .

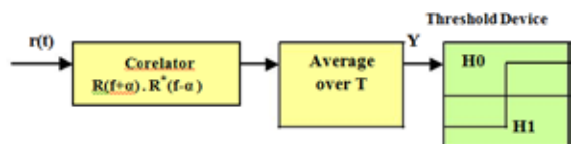


Fig. 9. Block diagram of feature detection

## V. COMPARISON OF VARIOUS SENSING TECHNIQUES

Comparison of different techniques of Primary Transmitter Detection is shown in figure 10. ED is the simplest one and also does not need prior knowledge of primary signal energy but ED needs long sensing time to achieve a given probability of detection. Its detection performance is also subject to the uncertainty of noise power and ED cannot be used to detect spread spectrum signals. Whereas Matched Filter detection needs less detection time because it requires few samples to meet a given probability of detection constraint but it requires a prior knowledge of every primary signal therefore if the information is not accurate, MF performs poorly. Also the most significant disadvantage of MF is that a CR would need a dedicated receiver for every type of primary user. In Cyclostationary detection method, its robustness to the uncertainty in noise conditions is its main advantage. Furthermore, it can distinguish the signals from different networks and also synchronization of CR with its neighbors is not required. Although feature detection is most effective method for spectral sensing but it is computationally complex and requires significantly long sensing time. Also its cyclostationary features may be completely lost due to channel fading [25].

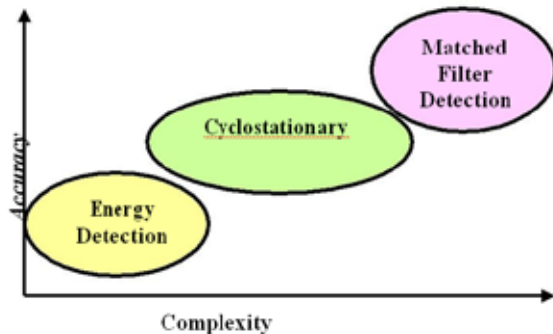


Fig. 10. Sensing accuracy and complexity of various sensing methods

## CONCLUSION AND FUTURE WORK

In this paper firstly we presented a survey of SDR Hardware architectures and Software platforms. Then a comparison of different technological choices available for implementing SDR like ASICs, FPGAs, GPP and DSPs is done. We found that FPGA is more advantageous as compared to other platforms because it provides reconfiguration, parallel processing, flexible memory structures, parallel and pipelined dataflow, flexible I/O and high speed. Next, we introduced Cognitive radio, which utilizes the available spectrum more efficiently through opportunistic spectrum usage. SDR provides an ideal platform for the realization of Cognitive radio. Finally we discussed and compared various aspects of Primary transmitter detection spectrum sensing techniques and we observed that cyclostationary-based methods perform worse than energy detector based sensing methods when the noise is stationary. However, in case of non-stationary noise energy detector based schemes fail while cyclostationarity-based algorithms are not affected.

Estimation of spectrum usage in multiple dimensions including time, frequency, space, angle, and code; identifying opportunities in these dimensions; and developing algorithms for prediction into the future using past information can be considered as some of the open research areas.

## REFERENCE

- [1] J. Mitola III, "Software radios: Survey, critical evaluation and future directions," IEEE Aerospace and Electronic Systems Magazine, vol. 8, pp. 25-36, Apr. 1993.
- [2] R. I. Lackey and D. W. Upmal, "SPEAKeasy: The military software radio," IEEE Communications Magazine, vol. 33, pp. 51-61, May 1995.
- [3] J. Mitola, "The software radio architecture," IEEE Communications Magazine, pp. 26-38, May 1995.
- [4] J. Mitola III, "Software radio architecture: a mathematical perspective," IEEE Journal on Selected Areas in Communication, vol. 17, no. 4, April 1999.
- [5] J. Mitola III, "Cognitive radio for flexible mobile multimedia communications," in Proceedings of the IEEE International Workshop on Mobile Multimedia Communications, San Diego, CA, USA, vol. 1, pp. 3-10, Nov. 1999.
- [6] J. Mitola III, "Cognitive radio: An integrated agent architecture for software defined radio." PhD thesis, Royal Institute of Technology (KTH), Stockholm, Sweden, May 2000.
- [7] Enrico Buracchini, "The software radio concept," IEEE Communications Magazine, vol. 38, no. 9, Sept. 2000.
- [8] Parul Gupta, Shivkumar Kalyanaraman, Pradipta Dey, Qing Wang, Jian Wen Chen, Yong Hua Lin "Software Radio: A Review of Design Considerations and Digital Hardware Choices", IBM Research Lab.
- [9] Ahmad Ali Tabassam, Farhan Azmat Ali, Sumit Kalsait2, Muhammad Uzair Suleman, "Building Software-Defined Radios in MATLAB Simulink - A Step Towards Cognitive Radios," IEEE 13th International Conference on Modelling and Simulation (UKSim), pp. 492-497, March 30 - April 1 2011.
- [10] C. Clark, Software Defined Radio: with GNU Radio and USRP, McGraw-Hill Professional, Nov. 2008.
- [11] P. Murphy, A. Sabharwal, and B. Aazhang, "Design of WARP: Wireless open-access research platform," in Proc. of European Signal Processing Conference, June 2006.
- [12] C. Chang, J. Wawrzyn, and R. W. Brodersen, "Bee2: a high-end reconfigurable computing system," in Proc. of IEEE Magazine, Design and Test of Computers, vol. 22, no. 2, April 2005.
- [13] G. J. Minden, et al, "KUAR: A flexible software defined radio development platform," in Proc. of IEEE Dynamic Spectrum Access Networks Symposium, April 2007.
- [14] L. Belanger, "Advanced SDR platform eases multiprotocol radio development," RF Design Magazine, January 2007.
- [15] H. Harada, "Software defined radio prototype toward cognitive radio communication systems," IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN-05), Maryland, Nov. 2005.
- [16] C. R. A. González et al., "Open-source SCA-based core framework and rapid development tools enable software-defined radio education and research," IEEE Communication Magazine, vol. 47, no. 10, pp. 48-55, Oct. 2009.
- [17] C. Hunter, J. Camp, P. Murphy, A. Sabharwal, and C. Dick, "A Flexible Framework for Wireless Medium Access Protocols," Invited Paper in Proc. of IEEE Signals, Systems and Computers Conference, Asilomar, Pacific Grove, CA, Nov. 2006.
- [18] T.R Newman, S.M.S. Hasan, D. Depoy, T. Bose, and J.H. Reed "Designing and deploying a building-wide cognitive radio network testbed," IEEE Communications Magazine, vol. 48, no 9, Sept. 2010.
- [19] I. Mitola, J. and J. Maguire, G. Q., "Cognitive radio: making software radios more personal," IEEE Personal Commun. Mag., vol. 6, no. 4, August 1999, pp. 13-18.
- [20] Federal Communications Commission, "Notice of proposed rule making and order: Facilitating opportunities for flexible, efficient, and reliable spectrum use employing cognitive radio technologies," ET Docket No. | 03-108, February 2005.
- [21] Takeshi Ikuma and Mort Naraghi-Pour (2008), "A Comparison of Three Classes of Spectrum Sensing Techniques", IEEE GLOBECOM proceedings.
- [22] Sabita maharjan, Kimtho Po, and Jun-ichi Takada, "Energy Detector Prototype for Cognitive Radio System," Technical report of IEICE.
- [23] R. Tandra and A. Sahai, "Fundamental limits on detection in low SNR under noise uncertainty," in Proc. IEEE Int. Conf. Wireless Networks, Commun. and Mobile Computing, vol. 1, Maui, HI, June 2005, pp. 464- 469.
- [24] Y. Hur, J. Park, W. Woo, J.S. Lee, K. Lim, C.-H. Lee, H.S. Kim, J. Laskar, "A cognitive radio (CR) system employing a dual-stage spectrum sensing technique: a multi-resolution spectrum sensing | (MRSS) and a temporal signature detection (TSD) technique," in Proc. | IEEE Globecom 2006, November 2006.
- [25] P. D. Sutton, J. Lotze, K. E. Nolan, and L. E. Doyle, "Cyclostationary signature detection in multipath rayleigh fading environments," in Proc IEEE Int. Conf. Cognitive Radio Oriented Wireless Networks and Commun. (Crowncom), Orlando, Florida, USA, Aug. 2007 |