RELIABLE AND ENERGY-EFFICIENT COGNITIVE RADIO COMMUNICATION

Dhivya S [1], Puja S [2], Suhasheni S [3], Priyadharshini G S [4]

[1]*Assistant Professor, Department of Electronics and Communication Engineering, Sri Ramakrishna Institute of Technology, Coimbatore, India*
[2,3,4] *UG Scholar, Department of Electronics and Communication Engineering,*
Sri Ramakrishna Institute of Technology, Coimbatore, India

---***---

Abstract - A crucial aspect in the world of communication is signal processing. Software Defined Radio makes it simple to comprehend and demonstrate real-time signal-processing ideas. The open-source programme GNU Radio, which serves as a simulation tool to drive the reception hardware, is the subject of this study. Here, RTL-SDR on/in GNU Radio is used to construct a wideband FM receiver. As a comparison to alternative gear, the suggested way of receiving signals using an RTL-SDR is more important and less expensive. Using RTL-SDR, Python, and GNU Radio, a low-cost experimental kit containing authentic radio signals from multiple sources is created. The information in this paper can be used to create CR systems that are trustworthy and usable in real-world settings.

Keywords – software-defined radio, GNU radio, RTL-SDR

1. INTRODUCTION

Most of the time, communication necessitates extensive cabling and wiring, such as that required by MOD-BUS, PROFI-BUS, CAN-BUS, Ethernet, etc.; this needs high installation and maintenance costs. To solve this, we use wireless technologies like Wireless Ethernet, etc., which do away with the need for cabling and reduce maintenance costs. If the choice is made to use wired or wireless communication, the decision-making process does not end there. Picking the best wireless technology is a significant choice. Software Defined Radio is one of these wireless technologies (SDR). SDR technology is used by more than 93% of the mobile infrastructure industry, and as demand for mobile data increases, more SDR base stations will be put in place. This project's primary impetus is the realization that by using With SDR technology, circuit complexity can be lowered while SNR is improved. By enabling the integration of radio functionalities into networking infrastructure as software modules that run on a generic hardware platform, SDR technology solves all of these issues. This eases migration from one generation to other as well as when travelling. SDR underwent a significant transformation in 2001 when compared to older radios because the functionality was primarily implemented in software. At that time, computers and data buses predominated over IF/RF in the SDR's circuitry. Two generalized SDR receiver techniques were suggested in 2005 research on digital quadrature transformation for Software Defined Radio (SDR) devices. One of those two can drop AD sample speed by two times, while the other does the opposite. With SDR technology, circuit complexity can be lowered while SNR is improved. SDR technology defeats all of these issues regarding AD sampling speed and output data rate. Modern radio communication applications utilizing SDR technology, such as radar, electronic warfare, and signal intelligence, evolved in 2012. Simulink, MATLAB, and Xilinx were able to create an SDR transceiver as a result. SDR is transforming wireless communication, which in 2013 became the industry to watch. To explore SDR, many open sources, like USRP (Universal Software Radio Peripheral) and GNU Radio, were frequently used. GNU Radio was mostly utilized as a simulation tool for the SDR transceiver hardware in 2015.

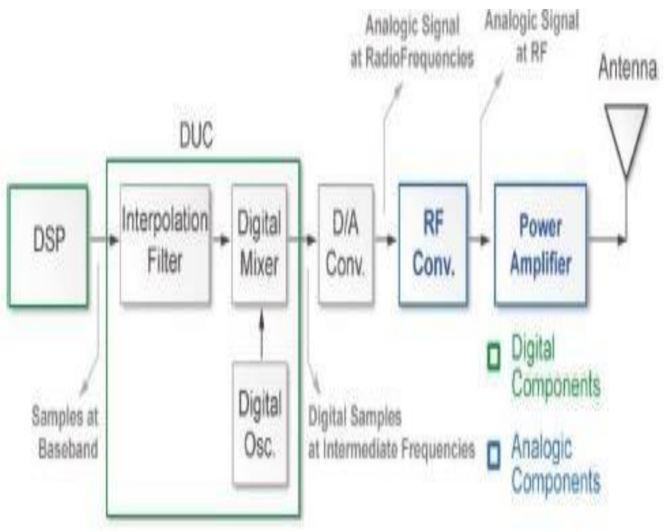
2. SOFTWARE-DEFINED RADIO HARDWARE

Software-defined radios are radio systems in which all physical layer or signal processing operations are carried out by software (SDR). We define the functions in software that are flexible and reconfigurable because if the function of any physical layer needs to be changed, then the hardware must be redesigned, which is more expensive. SDR development moves from military to commercial settings. The United States Navy developed Speakeasy, the first operational SDR, between 1991 and 1995. Hardware and software for SDR are both inexpensive. Because SDR technology allows radio equipment to be reprogrammed and reconfigured to function with numerous frequencies and signal types, they are significantly more adaptable and versatile than traditional radios. SDR technology also makes it easier to update and upgrade radio devices because software modifications may be performed rather than needing to replace actual hardware components. Technology has a wide range of uses, including in scientific research, military and defense, amateur radio, and wireless communications.

2.1 SDR TRANSMITTER

The transmitter is shown in Fig. 1. First, the DUC (Digital Up Converter) block converts baseband signals to IF, and then the DAC block converts samples to the analogue domain. Now, an RF converter shifts the signal to a higher frequency. The signal is finally boosted before being delivered to the antenna for broadcast. The baseband signal sampling rate in DUC is increased to match the operational frequency via the interpolation filter. Lastly, a digital mixer and local oscillator are used to shift the IF samples. According to the SDR transmitter block diagram, an SDR transmitter's essential parts include an antenna, digital signal processing, digital-to-analogue conversion, RF up-conversion, and power amplification. Each of these functional blocks can be processed differently to adapt to various communication standards and frequencies thanks to the flexibility of SDR technology.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 03 | March - 2023 Impact Factor: 7.185 ISSN: 2582-3930



2.2 SDR RECEIVER

The ADC converter receives the IF signal after that and is responsible for modifying the signal's domain so that its output offers digital samples. The samples are fed into the input of the digital down converter stage after that (DDC). The DDC, which is frequently a monolithic chip, is the essential component of the SDR system. A digital mixer, a digital local oscillator, and a Finite Impulse Response (FIR) low-pass filter make up its three primary parts. Similar to their analogue counterparts, the components operate. The IF digital samples are converted to baseband by the digital mixer and local oscillator, and the final signal's bandwidth is constrained by the FIR low-pass filter. The intended signal can be moved away from or towards the point when it reaches 0Hz by altering the tuning of the digital local oscillator. Which portion of the reception is recognised as a meaningful signal depends on this variation and the low-pass filter's bandwidth adjustment.

2.3 RTL2831 DEVICE

The SDR device will now be introduced after the structure of the SDR receiver and transmitter has been described. The Teratec RTL2831SDR receiver, one of the most affordable options on the market, is a great option for a first introduction to the technology. It uses the VHF and UHF frequencies to function, enabling the study of a sizable portion of the spectrum used for national broadcasts in a variety of applications. It provides a 3,2 MHz spectral width for the real-time operation of the DSP stage. The RTL2831 has a relatively tiny antenna that can be adjusted from 9 to 32 cm, however, it can be connected to additional antennas with superior performance that are suited to the operation's target band. Moreover, the gadget contains a USB 2.0 connector for connecting to a computer that is appropriate for the spectral width it can handle. Typical network cables are used to link devices with increased bandwidth monitoring capabilities. an extremely affordable, simple-to-use USB radio signal receiver. They were initially intended to be a DVB-T (Digital Video Broadcast-Terrestrial) receiver, but it was later discovered that by just switching the modes of the USBs, they could also be used as general SDRs. Any RF signal tuned by the tuner can be received by these SDRs. According to the components used, the USB might receive a different frequency range depending on the device. Yet the common frequency range is from 25MHz to 1.75GHz. In RTL-SDR, the antenna is quadrature down converted after receiving RF signals. These in-phase and quadrature-phase samples are transferred to the MATLAB-running computer or personal computer. The receiver design is put into practice using DSP techniques, which also demodulate the RF signal into a baseband signal. In addition to FM radio transmissions, this RTL-SDR can also receive signals from UHF/DTV, Digital Audio Broadcast (DAB) radio, 2G, 3G, and 4G cellular networks, GPS signals, scientific and medical (ISM) bands, and more. In actuality, RTLSDR may pick up any signal within the tuner's range.

2.4 SDR SOFTWARE

While the SDR concept depends on the hardware elements, the formulation of the paradigm itself emphasizes the need for complementary specialized software. The primary software tools that enable the manipulation of SDR signals are described in this section.

2.5 SDR FRAMEWORK

Software is required to enable the interface to operate an SDR device from a personal computer or an FPGA running Digital Signal Processing. But first, a framework that provides low-level interface functions needs to be built.

2.6 SDR USES

One can start looking for applications of the technology and providing specific solutions after the SDR device is in communication with the personal computer. The concept of one platform and the capacity to fix faults in real time are the classic applications of SDR. Several potential applications, however, have been suggested by research, including dynamic spectrum positioning, opportunity-driven multiple access (ODMA), spectrum regulation, and cost reduction (some SDR implementations are cheaper than their analogue counterpart) The SDR idea is starting to make an impact in high-impact telecommunications fields, a little outside its conventional uses**.** For example, HF Propagation Analysis, Interpretation of Cellular Technology Emissions, notably the OFDM modulation, and Identification of Radio Frequency Emissions are cases where this is applicable. SDR tests have produced good outcomes in various imaginative sectors, which encourages researchers to carry out more research. Numerous fields, including aviation tests, evaluation of multi-path communications, broadcast transmissions in multi-media mobile environments, cooperative wireless networks diversity, crossings prototypes between wireless networks layer, quantum optical communications, and especially cognitive radio research, are showing signs of having potential applications.

2.7 SDR SHARP

The first programme, SDR-Sharp, displays all of the readings that the SDR device, which translates to 3.2 MHz in the instance of the Teratec RTL2831, is capable of producing in real time. Three FM radio stations are visible in the chosen example's top spectrum, which is exhibited in real-time. The next window, referred to as a waterfall chart, depicts the signal's time behaviour by displaying the strongest emissions in warmer tones. Plotting the chosen bandwidth within the entirety of the spectrum shown in the top window is the responsibility of the two lower windows. The IF spectrum may be seen to the left. The voice-demodulated signal's frequency spectrum is shown in the illustration to the right. Also, if a speaker is attached to the computer, the acoustic content of the transmission can be heard. In the aforementioned scenario, an FM demodulator was undoubtedly employed in the demodulation process. Demodulation of the AM (Amplitude Modulation), CW (Continuous Wave), USB (Upper Side Band), LSB (Lower Side Band), and DSB (Double Side Band) signals is also possible using the programme. It is possible to identify some applications by comprehending how SDR-Sharp functions. They are displayed below:

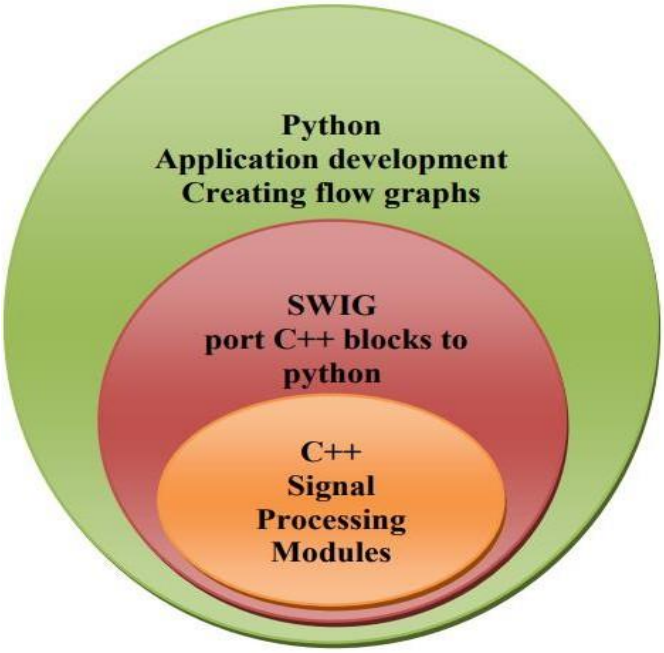
- Cheap Radio Receiver
- Spectrum Relocation
- Checking Repeaters Systems

2.8 GNU RADIO

A software radio can be implemented using the free source GNU Radio toolkit, which provides readily available signal processing modules. It can be used to build software-defined radios using generally accessible, inexpensive RF hardware. To support wireless communication researchers and actual radio systems, it is most frequently employed in academia and commercial settings. The original GNU Radio project was started by Eric Blossom. Major Linux distributions are supported by GNU Radio, and the packages are pre-compiled. Limited functionality is also developed for a port to Windows. The signal processing building blocks necessary to build radios, including modulators, demodulators, filters, etc., are included in the GNU Radio library. Critical signal processing pieces of GNU Radio are implemented in C++, and higher-level structuring, linking, and glueing of the blocks are carried out using Python. Moreover, GNU Radio Companion, a graphical environment for building bespoke radios, is provided (GRC).

2.9 GNU RADIO ARCHITECTURE

The architecture of the GNU radio software is made up of intricate flow graphs with a module and simple algorithms. These C++ modules and low-level algorithms offer fundamental signal-processing features (ex: filters, FFT, channel encoding etc ;). With the help of the Python "wrapper" or interface SWIG (Simplified Wrapper and Interface Generator), which also compiles and permits the integration between C++ and Python, these blocks/modules are automatically generated into python modules. The signal processing modules are typically written in C++, and python scripting is used to connect these blocks to create a flow graph. A flow graph model is built using these components, which are produced using Python.



SOURCE: Here is where the flow graph begins. There is a signal source in each flow graph. The most popular signal sources include USRP sources, file sources, etc.

FLOW GRAPH: Every application depends on a flow graph, where each flow graph is made up of blocks that act as intermediary sources and sinks. Several flow graphs might exist in a single application.

SCHEDULER: To develop an active flow graph, which is in charge of transmitting data through flow graphs, it is based on the data flow between the blocks. The scheduler checks the input and output buffers to see if there is enough data to start processing each block's processing functions. Linux, Mac OS X, and NetBSD are just a few of the operating systems that GNU Radio may run on. GNU Radio Companion is a graphical interface that enables users to quickly develop GNU Radio apps. Each application may be added to a list of accessible modules by simply double-clicking or dragging. It is quite simple to connect modules, and once the blocks are connected and moved, GRC immediately creates the necessary Python code and launches the programme.

2.10 PYTHON

The suggested framework is implemented using Python, an object-oriented high-level programming language. Python was chosen mostly because it has fewer lines of code and is simpler for users to understand than other programming languages. Several programming languages, including Java, C++, and C#, can also be used to create this code. Python may be used on a wide variety of platforms, and because its scripts are run on python interpreters, which can be found on both desktop and mobile operating systems, they are independent of the operating system on which they are installed.

2.11 IMPLEMENTATION

Even though commercial FM radio stations broadcast globally, the quality of the received signal varies greatly depending on where you are. Compared to AM systems, FM systems can readily reduce noise. Wideband FM (WBFM), the commercial standard used by all radio stations, is what we mean when we refer to "FM radio" in this context. This has a 75KHz frequency variation. There are countless sidebands in an FM broadcast. Every positive and negative multiple of the information signal surrounding the carrier has one of these sidebands. The WBFM receiver configuration for our project includes the following components:

• An RTL-SDR dongle and a stock antenna.

• A modern laptop computer running Linux.

• Supporting software packages, including,

1. SDR – It is a PC-based application to provide real-time radio functionality, and data recording and it fully supports RTL-SDR devices.

2. GNU Radio – It is accessible from a bootable Ubuntu Linux Live USB flash drive.

2.12 HARDWARE REQUIREMENTS

a) laptop containing the following specifications

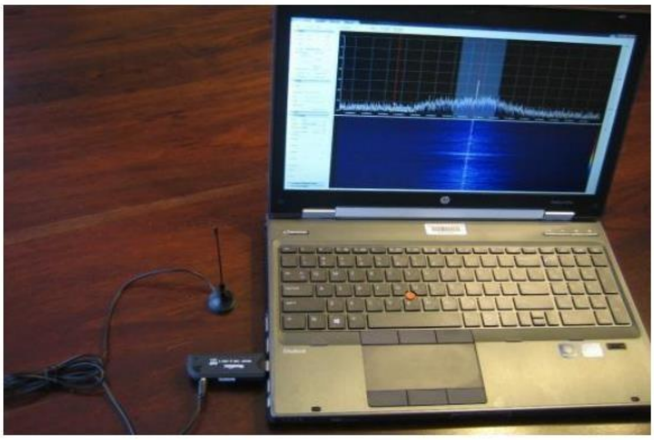
- Core i3 – processor or above
- 4GB RAM
- Hard disk – 250GB (minimum)

b) RTL-SDR dongle with an antenna (RTL382U R820T)

2.13 SOFTWARE REQUIREMENTS

a. Linux OS: Ubuntu 14.04

b. GNU Radio 3.6.4.1



A computer running GNU Radio Companion is connected to the R820T dongle and antenna, which is then coupled with the antenna in a WBFM receiver arrangement. Directly picked up from the air by the antenna, the FM signal is then processed in the computer before being demodulated. With a sound card, this demodulated signal is output.

2.14 RESULTS AND DISCUSSION

In this section, the results obtained by the above implementation are given. The output obtained at every stage is given here. These results are projected in the form of a time plot. The time plot gives the variation of amplitude with time.

3. CONCLUSION

The current wireless communication infrastructure will undergo a technical revolution thanks to Software Defined Radio. GNU Radio is a strong Software Defined Radio platform that may be used to build a digital signal processing chain while saving time and money. In our work, we combine the well-known frequency demodulation technique used in wireless communication systems with SDR to create a novel idea. The best hardware option, with a sufficient frequency range and BW for instructional purposes, appears to be an RTL-SDR dongle. Python's simplicity and scientific computing packages make it a popular programming environment. RTL-SDR has greater versatility than other methods. The benefit of this project is that building a receiver doesn't require writing a single line of code. SDR makes it easier for us to inspect the spectrum, detect interference, find unauthorized users of the spectrum, and categorize noise according to bands. This continued expansion of SDR with global receiving sites demonstrates the creation of a massive global network where it is possible to use the internet to listen to a radio broadcast from any location in the globe.

REFERENCES

[1] "WIPRO", 21st January 2012 [online] Available: http//:www.broadcastpapers.com/broadband/WiproSDRadio.pdf.[Accessed 25 April 2013].

[2] A. L. Reis, A. F. Selva, K. G. Lenzi, S. E. Barbin and L. G. Meloni,"Software Defined Radio on Digital Communications: A new Teaching Tool," in Wireless and Microwave technology conference, 2012 IEEE 13th Annual. IEEE, 2012, pp.1-8.

[3] Rao Arvind Malleri, Khyati Vachhani, 2015 IEEE "Experimental Study on Wideband FM Receiver using GNU Radio and RTL-SDR".

[4] J. Mitola III, "Software Radios. The survey, Critical Evaluation and Future Directions", IEEE National Telesystems Conference, pp. 13-15, 1992.

[5] R. J. Lackey and D. W. Upmal, "Speakeasy: The Military Software Radio", IEEE Communications Magazine, vol.33, pp.56-61, 1995.

[6] (August 2014). SDR-Sharp official Website. Available: http://sdrdharp.com

[7] (August 2014). Amazon. Venta de Hardware SDR. Available: http://www.amazon.com

[8] V. Giannini, J. Craninckx, and A. Baschirotto, "Baseband Analog Circuits for Software Defined Radio", ed: Springer, 2008.

[9] B. Sklar, "Digital Communications Fundamentals and Applications, ed: Prentice Hall, 2001.

[10] GNU Radio official website. http://gnuradio.org

[11] C++ language tutorial. http://www.cplusplus.com/doc/tutorial

[12] Python. Website: http://www.python.org

[13] Alex Verduin, "GNU Radio Wireless Protocol Analysis Approach", October 2008.

[14] BBN Technologies Corp. "GNU Radio Architectural Changes".

[15] S. Haykin, John Wiley & Sons," Communication Systems", 2008.

[16] V310c1r4pt0r.tk, "v310c1r4pt0r.tk using gnu radio companion simple fm radio tutorial," 2015. Available: http://v310c1r4pt0r.tk/2013/11/01/using-gnu-radio-companion-simplefm- radio-tutorial

[17] S. Cass.A $40 Software-Defined Radio.IEEE Spectrum, Vol. 50, No. 7, July 2013, pp. 22–23.

[18] Multi-Standard CMOS Terrestrial RF Tuner E4000 Datasheet. Elonics Ltd., 2010.

[19] R820T High-Performance Low Power Advanced Digital TV Silicon Tuner. Datasheet. Rafael Microelectronics, Inc., 2011.

[20] Communications System Toolbox Support Package for RTL-SDR Radio. User Guide. MathWorks, 2013.

[21] A. B. Sergienko. Possible Uses of USB Receivers Based on RTL2832U Demodulator for University Courses. Proc. 16th Int. Conf. on Digital Signal Processing and its Applications (DSPA-2014), Moscow, March 26–28, 2014, pp. 126–130. —in Russian.

[22] FCC, "Second Report and Order and Memorandum Opinion and Order in the Matter of Unlicensed Operation in the TV Broadcast Bands and Additional Spectrum for Unlicensed Devices Below 900 MHz and in the 3 GHz Band, ET. Docket No. 08-260," November 2008.