Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 01, January-2024

Software Defined Radio using NI LabVIEW

Dr.Pavithra A C
Asscociate Professor
Department of CSD
ATME College of Engineering Mysuru, Karnataka, India

Manjunath K
Assistant Professor
Department of ECE
ATME College of Engineering Mysuru, Karnataka, India

Juslin F
Assistant Professor
Department of ECE
ATME College of Engineering Mysuru, Karnataka, India

Abstract— **The Universal Software Radio Peripheral (USRP) The Universal Software Radio Peripheral (USRP) has become widely popular among Universities for hardware-based research and testing in the fields of Software Defined Radio (SDR) and Cognitive Radio (CR). The USRP now offers a scalable, simplified, and user-friendly unified platform with the most recent version of National Instruments (NI) LabVIEW. The introduction of new software support to the USRP platform expands its usability for educational purposes and will promote its further acceptance in teaching laboratories, university communication systems classrooms, and their natural follow-on coursework. This paper explores how LabVIEW based virtual instrumentation can be utilized with the USRP to quickly develop real-time communication system demonstrations for classroom and laboratory settings. The integration of the USRP, LabVIEW, and Windows support facilitates the implementation and exploration of both fundamental and advanced concepts in signal processing and communications**.

I INTRODUCTION

SDR is a technology that uses software modules to implement radio functionalities on a generic hardware platform. Integrating NI USRP hardware with LabVIEW software to create a versatile and powerful platform for rapid prototyping that supports physical layer design, wireless signal recording and playback, algorithm validation, signal intelligence, and various other advanced applications.

The majority of USRPs include a high-speed link to a host computer, which is used by the host-based software to communicate with the USRP hardware and send and receive data. In order to enable the USRP device to work independently, certain USRP versions incorporate an embedded CPU along with the standard features of a host computer.

LabVIEW includes a compiler that produces native code for the CPU platform. The graphical code is first converted into Dataflow Intermediate Representation, and then translated into segments of executable machine code by an LLVM-based compiler. The runtime engine executes these segments, enhancing overall performance. LabVIEW syntax is rigorously checked during editing and compiled into executable machine code either when running the program or saving it. In the latter case, the executable and source code are combined into a single binary file he execution of LabVIEW programs is managed by the LabVIEW run-time engine, which includes pre-compiled code to handle common tasks defined by the G language. This engine controls the execution flow and ensures a uniform interface with different operating systems, graphical systems, and hardware components. The run-time environment also enhances the portability of source code files across various supported platforms. LabVIEW programs generally run slower than equivalent compiled C code. However, as with other programming languages, optimizing the program can often help address and mitigate execution speed concerns.

II. LITERATURE SURVEY

SDR systems have a wide range of applications encompassing both digital and analog wireless communications. They are integral to wideband networks, spread spectrum technologies, navigation waveforms, emergency radio services, and public security systems. [1]. SDR systems can offer a wide range of communication services, depending on the waveform, design, and implementation. Faster transitions to more advanced digital signal processing methods and service-oriented applications are made possible by SDR technology, which reduces assessment and manufacturing costs [2].

The transmission and reception of radiation from the electromagnetic spectrum using radio waves with frequencies ranging from 30 KHz to 300 GHz is known as radio communication [3]. SDR is wireless equipment that uses programming to process radio waves instead of the more common analog hardware parts including detectors, modulators, amplifiers, filters, and mixers [4]. Since SDR technology can transmit and receive signals at a wide range of frequencies, it makes it possible to deploy wireless communication systems like FM radio, WLAN, LTE, and 5G [5].

The digital domain specifies the radio's operations, and software functions can be adjusted to alter the system. Additional communication system enhancements and the system's dynamic characteristics can be customized for the software-defined radio architecture. This technique is applicable to any modern radio-frequency (RF) communication device, such as emergency radios, cellular base stations, and defence communications systems. [6].

III. PROPOSED METHODOLOGY

The simplified overview of a SDR setup built around an NI US is shown in fig 1.1.The figure constitutes RF Transceiver and Software Processing. The function palette contains Transmitter- receiver, USRP properties synchronization, utility and examples

Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 01, January-2024

Fig 1.1: Simplified Overview of a SDR Setup Built Around an NI US

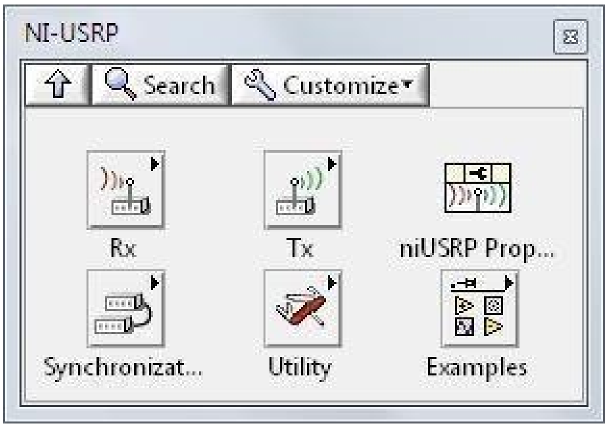

Fig 1.2:NI USRP function palette

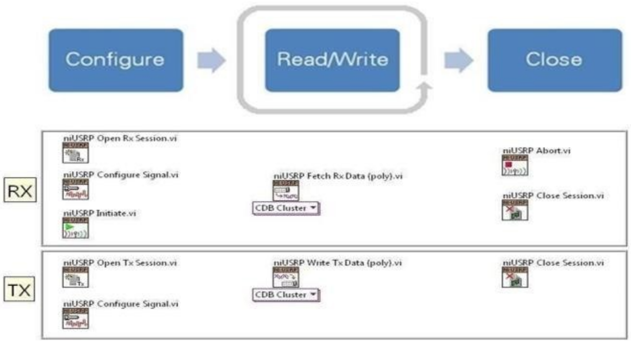

Fig1.3: NI-USRP Functions

NI USRP OPEN RX SESSION

The NIUSRP Open Rx Session VI is the initial VI utilized to establish a software session with the USRP for receiving RF signals. This session is essential for transmitting configuration settings and acquiring IQ data from the USRP. Notably, an Rx session is dedicated exclusively to functions related to signal reception.

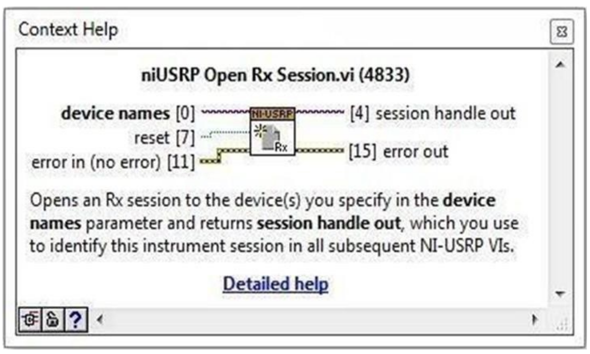

Fig1.4: The NI-USRP VI for Rx Function

The NIUSRP Configure Signal VI is compatible with both receive (Rx) and transmit (Tx) sessions. It adjusts parameters such as IQ rate, carrier frequency, gain, and active antenna settings.

When configuring multiple USRP setups, the channel list designates a particular USRP device. It's important to note that not all IQ rates, frequencies, and gain settings may be applicable or supported.Always read the coerced values to see if the requested and actual (coerced) values are different.

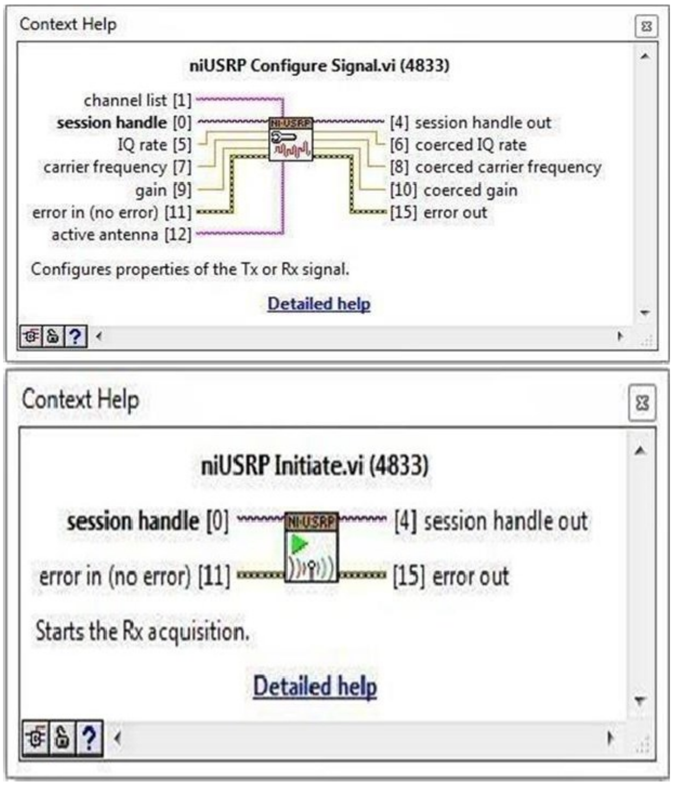

Fig1.5: NIUSRP VI for Configure Signal and initiate

NI USRP OPEN TX SESSION

The NIUSRP Open TX Session VI is the initial VI used to establish a connection with the USRP for transmitting RF signals. This session is crucial for sending configuration settings and transmitting IQ data to the USRP. A TX session is exclusively dedicated to transmission functions.

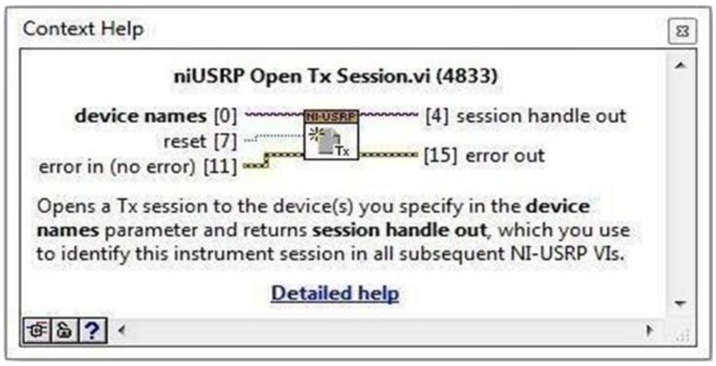

Fig1.6: NIUSRP VI for Open TX Session

Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 01, January-2024

FETCH RECEIVE DATA

The NIUSRP Fetch RX Data VI enables the retrieval of IQ data from a USRP that has an active Rx session established using the NIUSRP Open Rx Session VI. This data can subsequently be visualized in the time domain or processed digitally for further analysis

This VI (Virtual Instrument) is polymorphic, which means it offers multiple versions or instances of the VI available for selection based on the data type intend to work with. This VI can only be used with an Rx session.

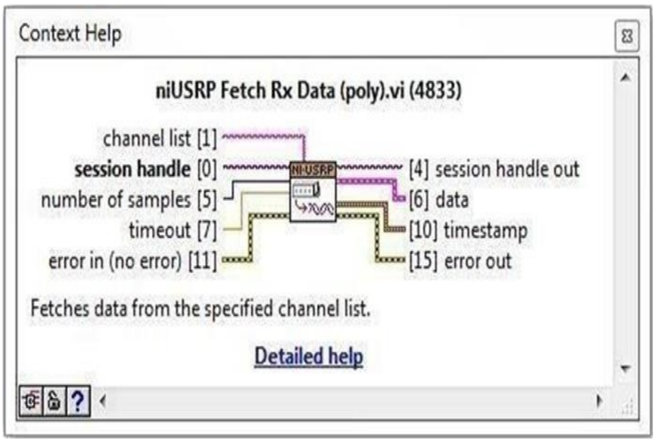

Fig1.7: NIUSRP VI to Fetch Rx Data

EXAMPLES OF USRP RECEIVE AND TRANSMIT

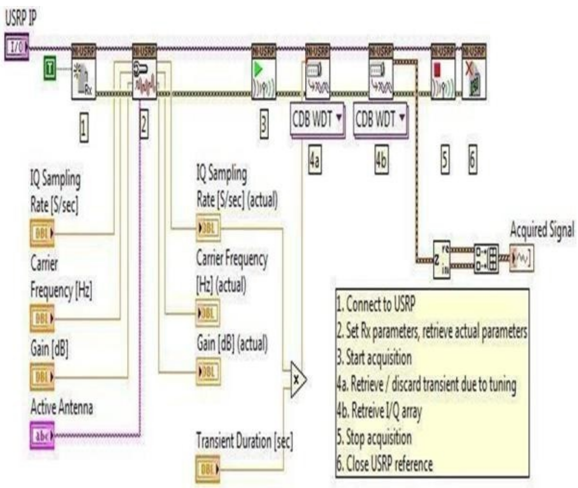

Fig1.8: USRP receiver and transmit examples

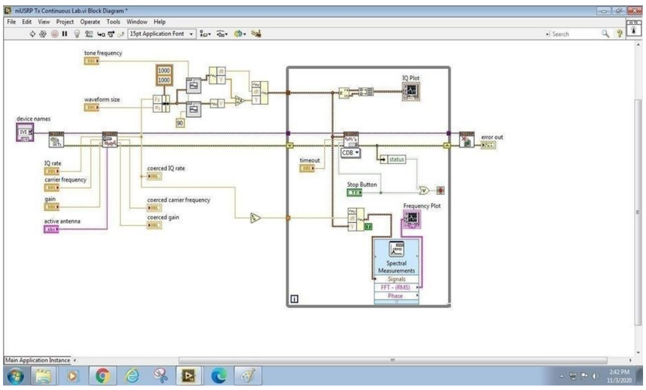

Fig 1.9: Continuous Transmission and Receiving of IQ signals

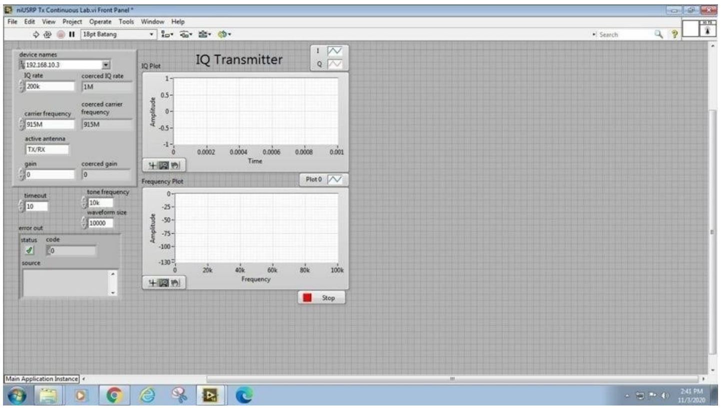

Fig 1.10: USRP Receiver Front Panel

TO TRANSMIT AND RECEIVE TEXT MESSAGES OVER VARIOUS MODULATION SCHEMES WITH NI USRP

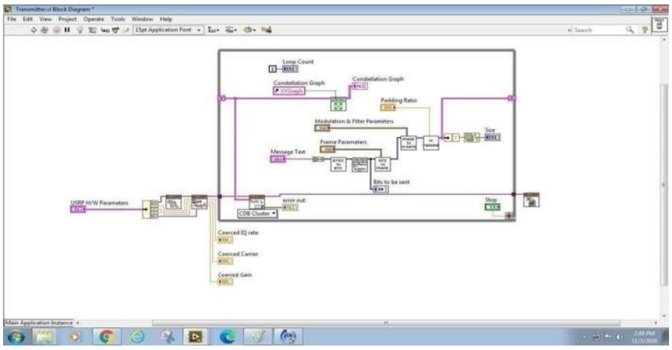

Fig1.11: USRP Transmitter

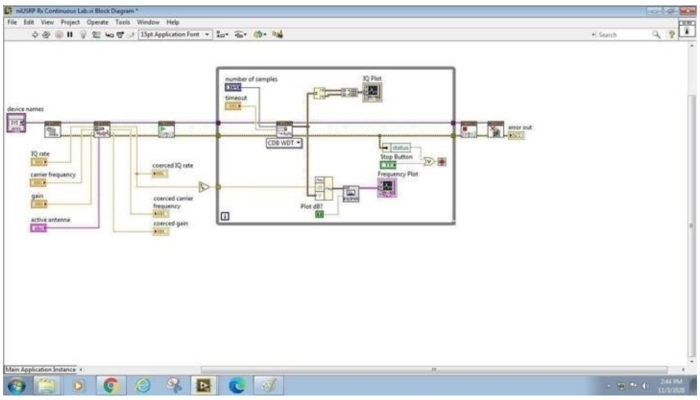

Fig1.12: Block Diagram of USRP Receiver

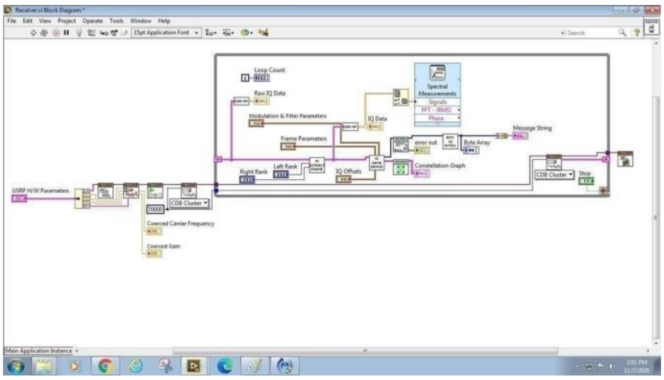

Fig 1.13: USRP Transmitter block diagram

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 01, January-2024

USRP RECEIVER

With graphical NI LabVIEW software, one can easily configure their network, collect measurement data, trigger alarms through SMS or e-mail, and even view monitoring data within a Web browser. It is the best tool used to develop automated research, validation, and production test systems. Learning the fundamentals of LabVIEW puts a powerful and very useful tool at your fingertips. It is easy to learn, has excellent documentation, and is the base for all modern control applications. Thus it is an instrumental software system and a competitive analysis tool used for measurement and automation.

Technically it's a development environment, and the language is "G", but in common usage it's a language. Instead of typing words like with C++, Python, or other text-based languages, you place and connect visual objects around your screen. The focus on improving project management starts with source code control tools to improve workflow. These controls will also be improved upon in future development. Developed work habits and attitudes necessary for job success. Develop communication, interpersonal and other critical skills in the job interview process. Build a record of work experience.

PROS, CONS AND APPLICATIONS PROS

- Integration with NI USRP Hardware LabVIEW integrates seamlessly with National Instruments (NI) USRP hardware, providing a robust platform for developing and deploying SDR applications.
- Graphical Programming: LabVIEW's graphical programming environment simplifies SDR development, making it accessible to engineers and researchers with varying programming backgrounds.
- Signal Processing Libraries: LabVIEW offers extensive signal processing libraries and modules, facilitating rapid algorithm development and testing.
- Scalability: LabVIEW supports scalable SDR implementations, from simple prototypes to complex multi-channel systems, leveraging its modular architecture.
- Visualization and Analysis Tools: LabVIEW provides powerful tools for signal visualization, analysis, and debugging, aiding in real-time system monitoring and optimization.

Cons:

- Performance: LabVIEW programs may run slower compared to equivalent code written in lower-level languages like C or C++. Optimization techniques are necessary to enhance performance for real-time applications.
- Learning Curve: While the graphical nature of LabVIEW is advantageous, mastering its programming paradigm and understanding dataflow can require time and effort.

Applications:

- Developing and evaluating the most recent cellular communication standards, including Long-Term Evolution (LTE) and 5G, for both uplink and downlink transmissions.
- Creating and assessing wireless protocols for instance Wi-Fi (802.11) and Bluetooth, ensuring they meet standards and optimizing their performance.

IV RESULT ANALYSIS

The development process is made simpler by LabVIEW's graphical programming environment, which offers visual tools for planning and implementing SDR systems. In comparison to conventional text-based programming languages, this can result in faster development and iteration, lowering the time-to-market for SDR applications. gear setup and configuration for SDR projects is made simpler by the seamless integration of LabVIEW with National Instruments (NI) USRP gear. Reliability and compatibility are improved by this integration, guaranteeing strong performance in a range of SDR applications. With the vast libraries and signal processing tools that LabVIEW provides, developers can create complex modulation/demodulation, filtering, and spectrum analysis algorithms. This capability improves system performance and flexibility by facilitating the implementation of advanced SDR functionalities.

V. CONCLUSIONS

In conclusion, LabVIEW stands as a robust tool with wide-ranging applications in the automation industry. In today's era of rapid technological advancement encompassing AI, virtual reality, robotics, and automated machinery, LabVIEW plays a crucial role in facilitating virtual automation's promising future. Projects developed with LabVIEW not only find industrial applications but also serve as effective simulations of real-world industrial processes.

VI . REFERENCES

[1] T. Chen, M. Matinmikko, X. Chen, X. Zhou and P. Ahokangas, "Software-defined mobile networks: concept, survey, and research directions." IEEE Communications Magazine, vol. 53, no. 11, pp. 126-133, 2015.
[2] H. Miyashiro, M. Medrano, J. Huarcaya and J. Lezama, "Software Defined Radio for hands-on Communication theory," in 2017 IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON), 2017.
[3] G.-D. Rogelio. P.-M. Ramón and S.-V.Jorge, "Design and verification of a testing platform for implementation of software-defined radio communications systems," Journal of Computational Systems and ICTs, vol. 5, no. 16, pp. 14-19, December 2019.
[4] I. T. Haque and N. Abu-Ghazaleh, " Wireless software-defined networking: A survey and taxonomy." IEEE Communications Surveys & Tutorials, vol. 18, no. 4, pp. 2713-2737, 2016.
[5] S. Mori and K. MizutaniI, "Software-Defined Radio- Based 5G Physical Layer Experimental Platforms For Highly Mobile Environments," IEEE Open Journal of Vehicle Technology, vol. 8, pp. 230-240, 2023.
[6] K.T. Foerster, S. Schmid and S. Vissicchio,"(2018). Survey of consistent software-defined network updates." IEEE Communications Surveys & Tutorials, vol. 21, no. 2, pp. 1435-1461, 2018.
[7] Raghunandan B H; Mahesh A; Mahesha Babu M.P. "Wireless Communication System Design Using Labview and Software Defined Radio". IEEE International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), DOI: 10.1109/CSITSS60515.2023 pp. 295-303, Nov 2023.
[8] L. Benetazzo, M. Bertocco, F. Ferraris, A. Ferrero, C. Offilli. M. Parvis, V. Piuri, "A web- based distributed virtual educational laboratory", IEEE Trans. Instr. Meas., vol.49,pp. 349-356, 2000.
[9] Beyon, J.Y., LabVIEW Programming, Data Acquisition and Analysis, Prentice Hall PTR (Upper Saddle River, NJ), 2001.
[10] Wells, L.K. and T. Jeffrey, LabVIEW for Everyone: Graphical Programming Made Even Easier, Prentice Hall PTR, (Upper Saddle River, NJ), 1996.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 01, January-2024

[11] Jamal, J. and H. Pichlik, LabVIEW Applications and Solution, Prentice Hall,(Upper Saddle River, NJ),1999.
[12] "Introduction to RC filter Lab" for Engr 60 at USD by one of the authors (SML) based on a lab by Thomas F. Schubert, Jr.Beams, D.M., "Project TUNA – The Development a LabVIEW Virtual Instrument as a Class Project in a Junior – Level Electronic Course" Session 22259, Proceedings of the 2000 ASEE Annual Conference , St. Louis, MO, June 2000.
[13] M.G. Guvench, S. Gile and S. Qazi, "Automated Measurement of Frequency Response of Electrical Networks Filters and Amplifires", Session 1359, Proceedings of the 2000 ASEE Annual Conference, St. Louis, MO, June 2000.
[14] A. Ferrero, V. Piuri, "A Simulation tool for Virtual Laboratory experiments in a www.environment",IEEE Trans. Instr. Meas., vol.48,pp. 741-746, 2020.
[15] Shujiao Ji and Ming Zhu, "The Simulation Design of Communication System Based on LabVIEW", IEEE 2nd International Conference on Measurement Information and Control, pp. 661-664, 16–18 Aug. 2013.
[16] Thad B. Welch and Sam Shearman, "Teaching Software Defined Radio Using The USRP And LabVIEW", IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), 2022
[17] Shujiao Ji and Ming Zhu, "The Simulation Design of Communication System Based on LabVIEW", IEEE 2nd International Conference on Measurement Information and Control, pp. 661-664, 16–18 Aug. 2013.
[18] Ian F. Akyildiz, Josep M. Jornet and Shuai Nie, "A New CubeSat Design with Reconfigurable Multi-Band Radios for Dynamic Spectrum Satellite Communication Networks", Ad Hoc Networks, 2018.
[19] I. T. Panagiotis et al., "Software Defined Radios for CubeSat Applications: A Brief Review and Methodology", IEEE Journal on Miniaturization for Air and Space Systems, 2020.
[20] S. Hemalatha, Sk. Imran et al., "Implementation of FSK Transceiver using Software Defined Radio (SDR)", International Research Journal of Engineering and Technology (IRJET), vol. 5, no. 3, March 2018.
[21] C. Zhiyong, "Universal CubeSat Platform Design Technique", MATEC Web of Conferences, vol. 179, no. 2018, pp. 01002, 2MAE 2018.
[22] Nasir Saeed et al., "CubeSat Communications: Recent Advances and Future Challenges", IEEE Communications Surveys& Tutorials, 2020.
[23] I. T. Panagiotis et al., "Software Defined Radios for CubeSat Applications: A Brief Review and Methodology", IEEE Journal on Miniaturization for Air and Space Systems, 2020.