Implementation of FM Transceiver using Software Defined Radio (SDR)

Devidas Kushnure, Murtaza Jiniyawala, Sushama Molawade, Snehal Patil

[1]Assistant Professor,[2]Student,[3]Student, [4]Student
Department of Electronics and Telecommunication Engineering,
VPKBIET, Baramati, India.

Abstract—In this Paper we discusses about recent trends in wireless Communication system design using Software Defined Radio (SDR). Here the Universal Software Radio Peripheral (USRP) B200 board is used to design Frequency Modulation (FM) Transmitter and Receiver. The audiofiletransmitted from the USRP B200 board and it is received on the FM receiver of mobile phone as well as on the USRP B200 board. The system has been designed using the GNU Radio which is open source software. In the demonstration the audio file is transmitted using Wideband Frequency Modulation (WBFM). The spectrum ofFM is from 88 MHz to 108 MHz within this spectrum wecan transmit and receive the FM signals. The results showthat the audio file transmitted is received on the FM Receiver of Mobile Phone as well as USRP B200 Board Receiver. The spectrum received is just like the transmitted spectrum.

Keywords: Software defined radio (SDR), GNU Radio,USRP, Wideband Frequency Modulation (WBFM).

I. INTRODUCTION

One of the most important discussions in current era is Software Defined Radio (SDR) which provides researchers with a powerful and flexible wireless communications experimentation platform. SDR will allow the programmer to realize various communication systems. SDR is digital programmable platform which has ability to realize the structure of the device with high mobility, reconfigurability and flexibility. Every SDR is comprised of software as well as hardware. In this paper, we consider GNU radio software coupled with universal software radio peripheral (USRP) hardware. SDR is radio communication system where elements have been implemented by software on personal computer. The practical research was very costly in terms of time as well as money. Both can be saved by using SDR. The translation of the signal processing into software which is run by GNU software on personal computer opens up a huge number of possibilities at an affordable price. Now we can change and set the parameters that were embedded in software and check out the final results. SDR will allow us to analyze and change every parameter in the system.
In this paper, we implement the system for transmission of the audio file and its reception on mobile phone using GNU radio software [1] [4] [7]. GNU radio provides the data flow abstraction and the fundamental concepts are signal processing blocks and connections between them.

II. BACKGROUND

A. GNU Radio

The GNU Radio is a set of software signal processingbuilding blocks that allows a user to create their ownsoftware radio [6]. It is free/open source software that offerslibrary of more than 1000 processing blocks to implementvarious software radio applications using a low cost externalRF hardware (USRP). The GNU radio allows variousreal time SDR applications to be performed on a singleUSRP board. The required signal processing and designingof various mixers, amplifiers, filters, spectrum analyzers,modulators, demodulators etc., are carried out with GNUradio on a PC/laptop using Linux [3]. It is mostly used foracademic and in the commercial environments to supportvarious wireless communication researches and to carry putreal world radio applications. Since GNU is a software toolit is capable of handling the digital data [1]. In GNU radiothe blocks can be created using high level programminglanguage Python and the high level signal processing blocksare created using C++. These modules are called fromPython using SWIG(Simplified Wrapper and Interface Generator)interface. In this paper the GNU schematics for FMtransmitter are shown.

B. Software Defined Radio (SDR)

Software Defined Radio (SDR) is defined as a radio platform that uses various software techniques on digitized radiosignals [2]. Therefore, in the process of receiving chain, besides flexibility, the main of SDR is to turn the hardware problems into software problems and allows working in more accessible domain [6]. SDR is divided into two subsystems: The hardware defined subsystem could consist of an antenna, an ADC and the second is Software Defined subsystem.For proper working, the device should satisfy the following three conditions:

1) The antenna should be capable to operate at the frequency of all radio signals [2]
2) ADC and DAC should operate at the sampling rate greater than twice the frequency of signal [2].
3) To handle the signal processing of all radio signals, the software subsystem should have enough processing power [2].

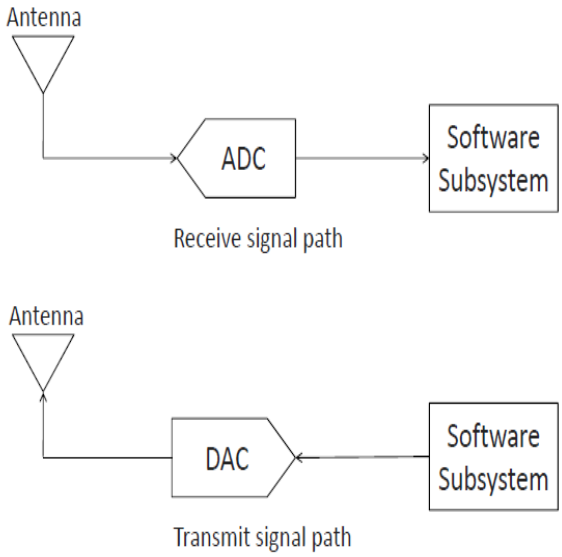

Fig 1: Block Diagram Representation of SDR

In SDR, hardware defined subsystem consists of wideband radio which takes the portion of the spectrum and converts it to IF before the digitization process [7]. The software defined subsystem receives the digitized wideband signal and then send to the down converters to isolate the carrier. Finally demodulation is performed. Programmable devices are combined to compose the software defined subsystem, such that connection between them is fast enough to transfer the DSP data [7].

C. USRP

Universal Software Peripheral (USRP) is Universal hardware that can be accessed using software which is GNU Radio software on LINUX platform [1]. Hardware mainly consists of three parts. First one is the radio frequency (RF) frontend which transmits and receive the radio signals. Second one acts as a bridge between analog block and digital block. All signal processing is done within the third block in software form.

The analog RF signal can be received or transmitted by antennas or directly connected via SMA connectors to the RF frontend called Daughter boards [8]. USRP has various daughterboards which are vary from each other depending on their operating frequency. These daughterboard are connecting to the motherboard and acts as RF frontend of USRP. The analog signals are converted to digital samples in USRP in motherboard and then downconverted within the FPGA. Sample rate is also varied by doing decimation. Then sampled data by the FPGA is sent to the host by USB. GNU Radio framework controls the further signal processing capabilities [8].

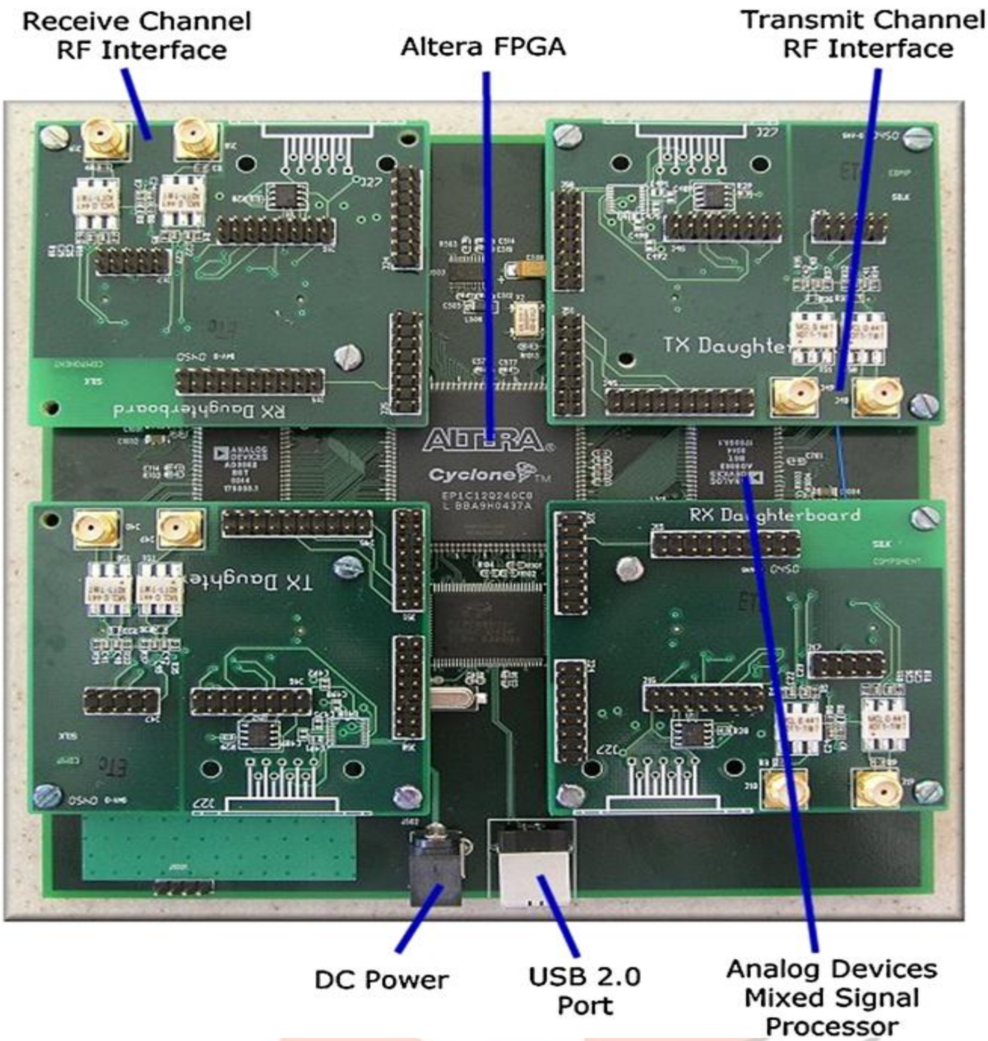

Fig 2: USRPB200 Board

III. IMPLEMENTATION

A. Implementation of FM Transmitter

In this section we are going to discuss the implementation of the FM Transmitter using Wideband Frequency Modulation Technique. In wideband frequency modulation technique the highest frequency component of a modulating signal is greater than that of the peak frequency deviation of modulated signal[4][7].

B. Frequency Modulation (FM)

In the signal processing and telecommunications, frequency modulation is encoding of the information in a carrier wave by varying the instantaneous frequency of the wave. In the FM radio broadcasting of an audio signal, representing the voice or music, the difference between the carrier frequency and its center frequency is proportional to that of the modulating signal [7]. Frequency Modulation is widely used in radio transmission because it has a larger signal-to-noise ratio and therefore it rejects radio frequency interference better that that of Amplitude Modulation signal. For this reason, the audio is broadcast over FM radio [4].

C. Implementation using GRC

In the implementation of the FM transmitter, the modulation technique used here is wideband frequency modulation. The fig 3 shows the detailed flowchart of the FM transceiver.

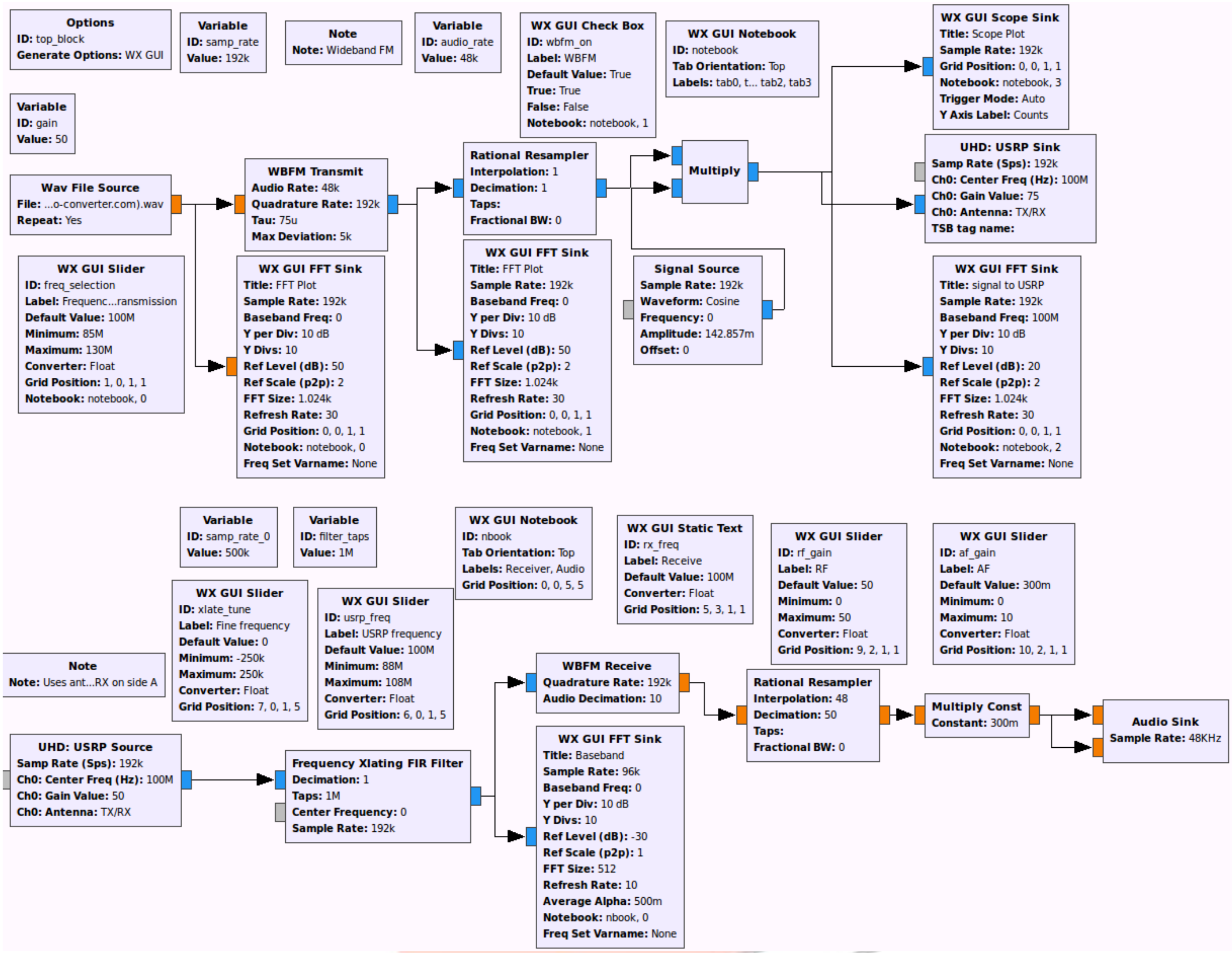

Fig 3: GRC Flowchart for FM Transmitter and Receiver

Here our aim is to transmit an audio file and receive it on a mobile phone and on same USRP Board. So we are going to use a file source which is a wav source file. In the Wav file source block we have to choose the input file source that is stored on our PC.

Fig 4 shows the graph for the Wav file before giving it to the WBFM Transmit block.

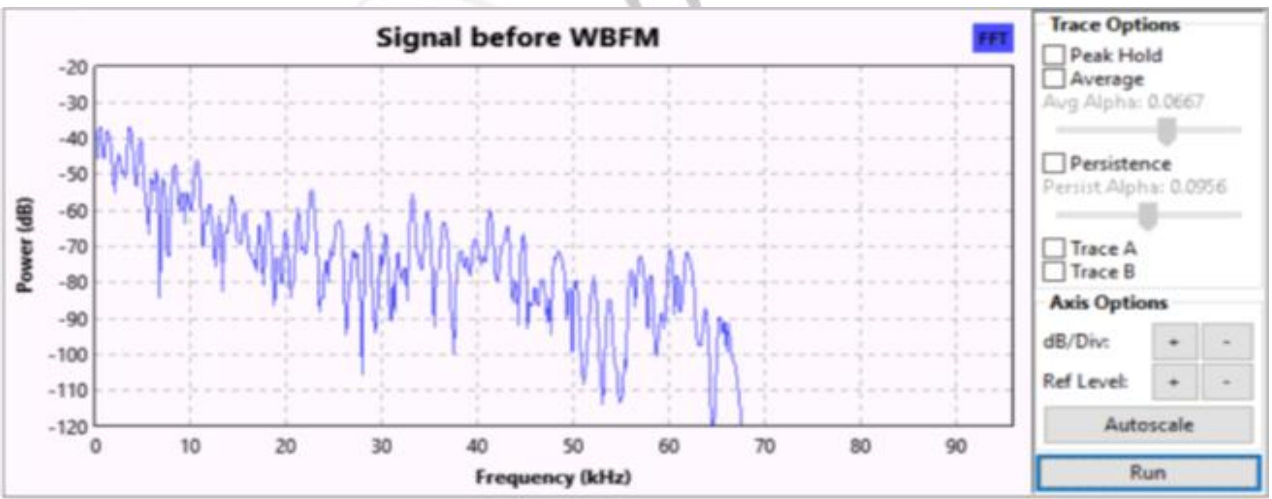

Fig 4: Signal before WBFM

Then this input wav file is sent to the WBFM Transmit block where the audio rate is predefined. Here quadrature rate is set in the multiples of 48; this rate is the outgoing sampling rate whereas 48 kHz is incoming sampling rate.

1. Audio rate: 48 kHz
2. Quadrature Rate: 48 kHz * 4 = 192 kHz
3. Tua: 75u

The modulated signal is shown in fig 5 below.

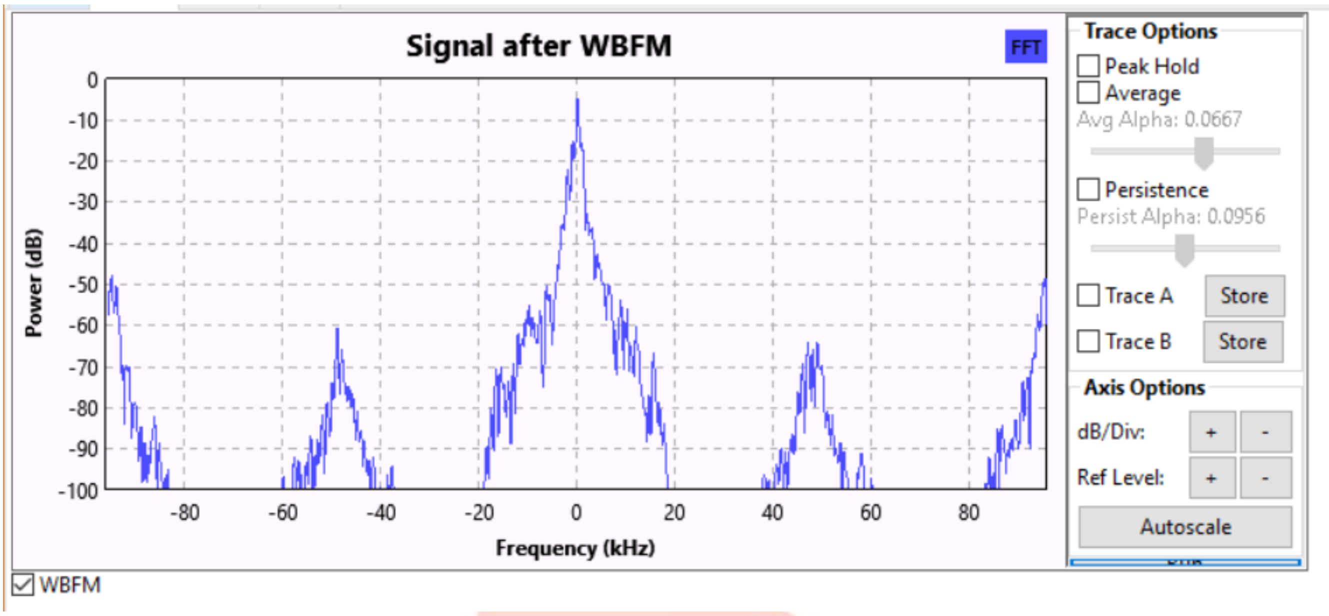

Fig 5: Signal after WBFM

Now, the sampled signal from the WBFM Transmit block is sent to the Rational Resampler block where we are doing some frequency conversion. For example, the Audio Sink requires lower frequency and USRP block requires the higher frequency then this block converts the frequency by doing some interpolation or decimation as per the requirements. The output of the Rational Resampler is given to one of the input of the multiply block and the second input of the multiply block is a cosine signal with a sample rate of 192 kHz. Here the carrier signal is used for proper transmission. Now these multiplied signal is given to the USRP sink from where these signals are send to the destination using wireless medium. Here we use 100 MHz as center frequency but it can be changed using freq_selection slider. The antenna is connected to TX/RX.

The fig 6 shows the FM broadcasting time domain plot

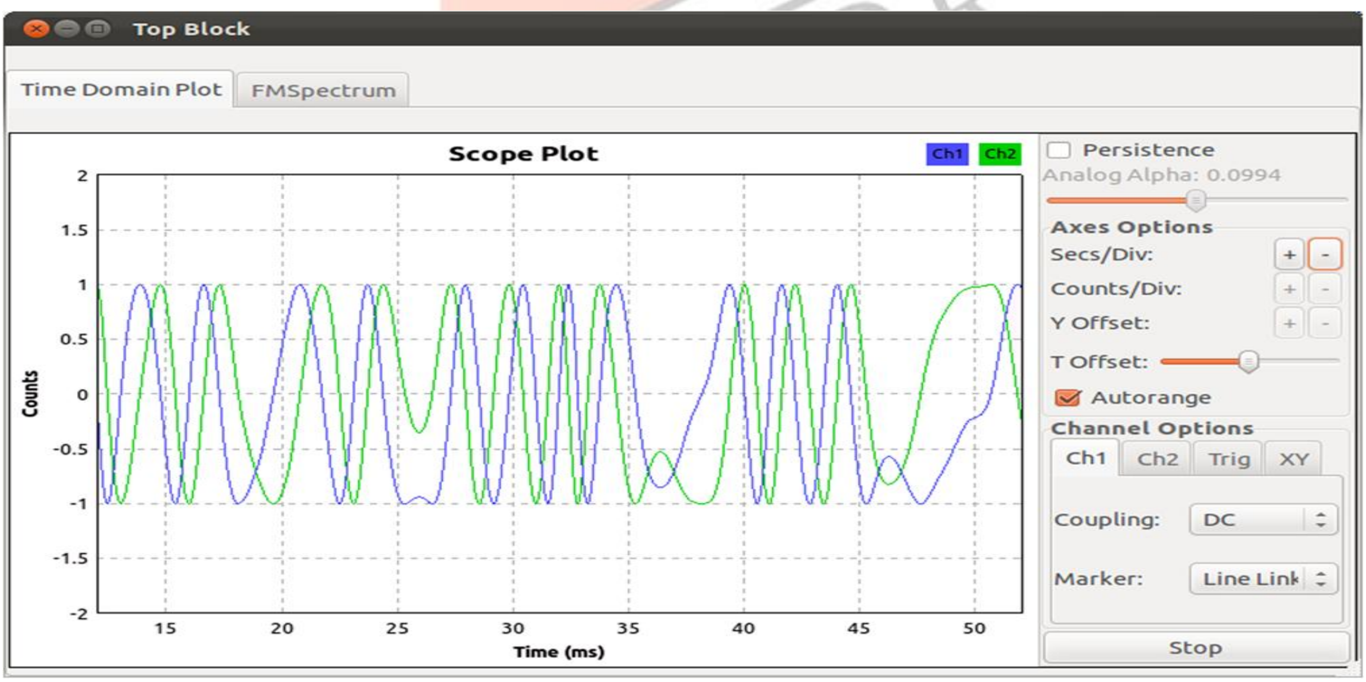

Fig 6: Time Domain Plot

The fig 7 shows the FFT plot of the output signal. The audio file is transmitted at the frequency of 100MHz.

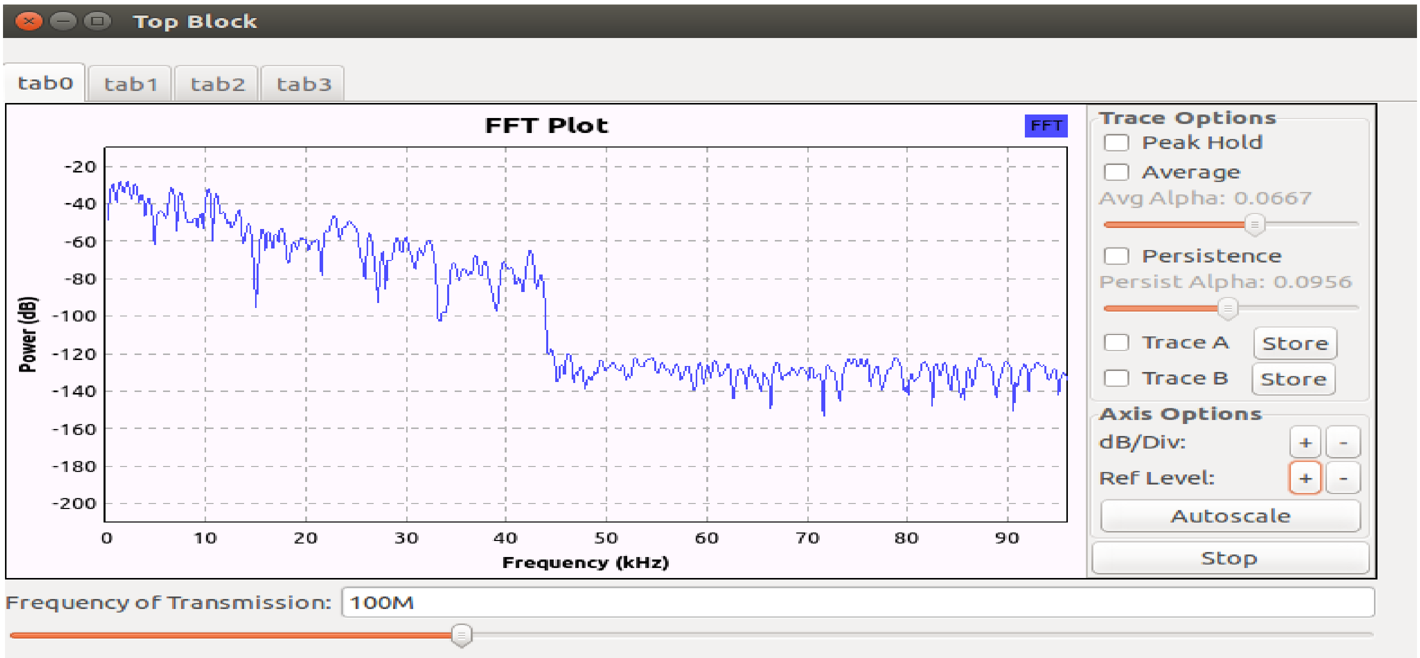

Fig 7: FFT Plot of the Output Signal

Now for the implementation of the Receiver, the input source taken is USRP block with the center frequency of 100MHz which can be adjusted by changing the parameters. Here the sampling rate is 192 KHz and the gain can be changed by the slider with min=0 and max=50. The antenna is connected to TX/RX in daughter board. Now the output of the USRP board is given to a filter. Here the filter passes the signal below the cutoff frequency and removes the higher frequency. Now the filtered output is given to the WBFM Receive block. This block demodulates the signal from the data stream and we get the original data stream that was send by the sender. Output of the WBFM block is send to the rational re-sampler block. Here some interpolation and decimation is done on the data stream according to the desired rate. Finally this signal is given to the Audio Sink for listening the sound of the receiver channel. The fig 8 shows the

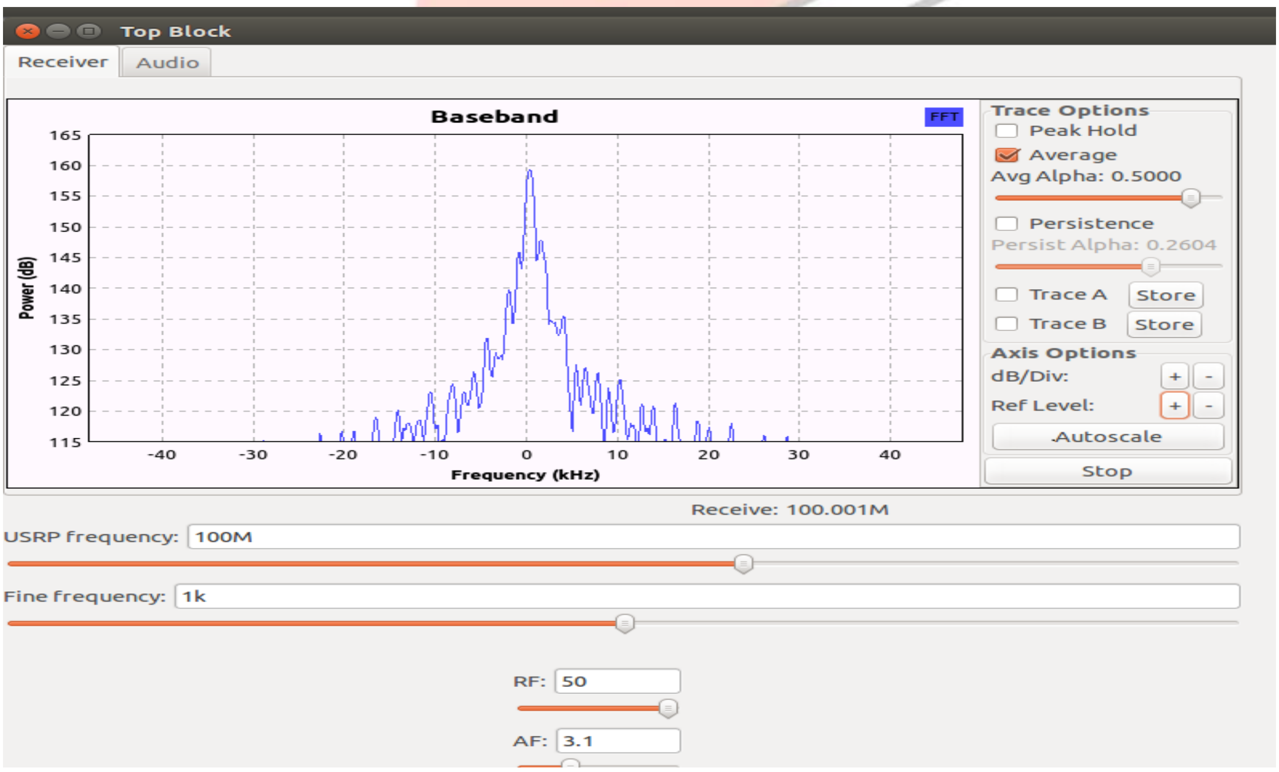

Fig 8: Receiver Output

IV. MOBILE TUNING

For listening the transmitted audio signal on the mobile phone we need to tune the mobile radio to the transmitting frequency that is 100MHz. Fig 9 shows the setup of the mobile phone.

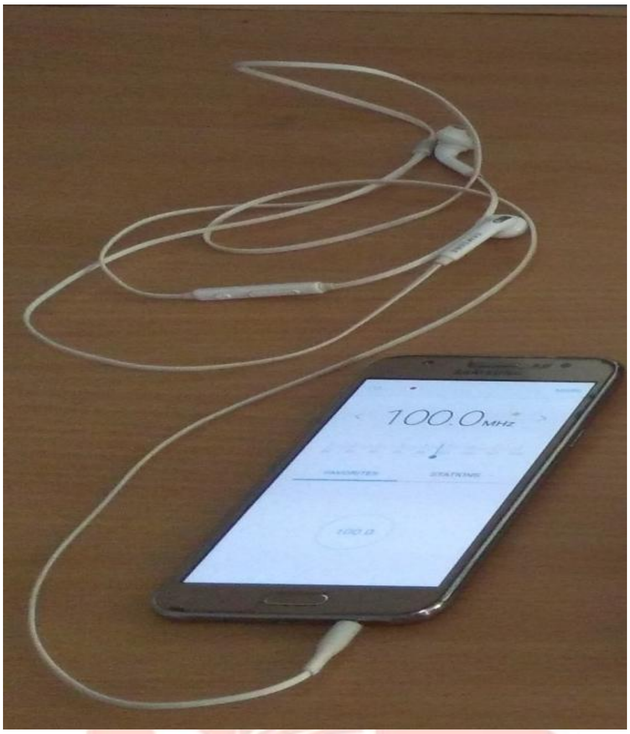

Fig 9: Mobile with FM Radio

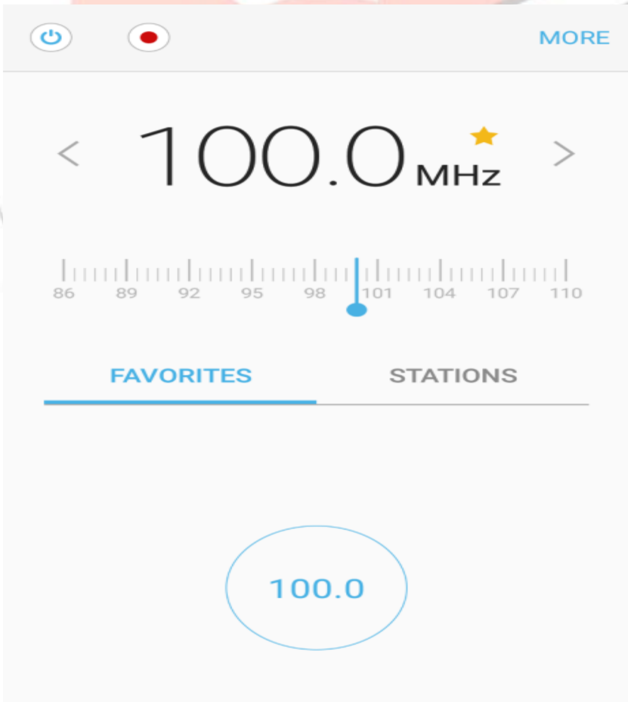

Fig 10: Tuning to 100MHz on Mobile FM

The fig.10 shows the tuning of the mobile phone to the 100MHz frequency which is our transmitting frequency. By tuning to this frequency we are able to hear the transmitted audio file.

V. HARDWARE SETUP

The fig 11 shows the complete setup of the Audio Transmitter and Receiver system along with the reception of the audio file on the mobile phone.

Fig 11: Experimental Setup

VI. CONCLUSION

In this paper, we demonstrated the usability of SDR to implement the communication system. The WBFM system implemented here gives the concept of Software Defined Radio with the help of USRPB200 as a hardware tool and GNU Radio as a software tool. We implemented a communication system with GNU radio software with USRP and we successfully transmit the audio file from USRP board and received it on the FM of mobile phone and USPR board. The audio can be received upto the distance of 100 meters clearly.

Hence it is clear that one can implement other communication systems using SDR and analyze the performance. Therefore Using SDR it is possible to design prototype of the Communication systems and possible to verify the real time performance of the system.

ACKNOWLEDGMENT

Authors would like to thank the Savitribai Phule Pune University, Pune for providing financial support to this project. We also would like to thank department of Electronics and Telecommunication VPKBIET, Baramati for providing necessary support.

REFERENCES

[1] Gandhiraj, R., and K.P.Soman"Modern analog and digital communication systems development using GNU Radio with USRP." Telecommunication Systems 56.3(2014): 367-381
[2] Ulversory Tore, "Software Defined Radio: Challenges and Opportunities" IEEE Communications Surveys and Tutorials 12.4 (2010): 531-550. IEEE. Web. 24 Mar. 2011
[3] Abinav, A. k., Naveena, K., Pratibha, R., Gandhiraj, R., & Soman," Detection and classification of signals to configure software defined radio", Second national conference on recent trends in communication, computation and signal processing, March 2627
[4] S. Aramvith and R. D. Cajote,"Handbook of Research on Secure Multimedia Distribution", ISBN-13: 978-1605662626 ,Pp. 211-240, Information Science Reference ,2009
[5] Z. Tong, M.S. Arifianto, C.F. Liau,"Wireless Transmission using Universal Software Radio Peripheral".Procending of the 2009 International Conference on Space Science and Communication, Port Dickson, Negeri Sembilan, Malaysia, 26-27 October 2009.
[6] Dillinger, M., Madani, K., & Alonistioti, N. (2005). Software defined radio basics. Review of Software Defined Radio: Architectures, 6(10).

[7] H. Chang, H. Jin, X. Zhang, D. Liang, M. Peng, and W. Wang, "ASoftware Defined Radio Scheme for the Hierarchical Modulation Based Multimedia Broadcast System Using USRP," 6th Wireless Communications and Mobile Computing (WiCOM 2010), pp. 1-4., 2010.
[8] Blossom, Eric, Johnathan Corgan, Matt Ettus, and Tom Rondeau. GNU Radio.2006. Web. 27 Feb. 2011
[9] Brannon, Brad, "Software Defined Radio Analog Devices", INC.,2008.Web.24 Mar. 2011
[10] Proakis, J. G., & Salehi, M. (2007). Fundamentals of communication systems. Pearson Prentice Hall, India.
[11] GNU radio introduction. http://gnuradio.org/redmine/projects/ gnuradio/wiki/WhatIsGR
[12] GNU radio companion. http://www.gnuradio.org/trac/wiki GNURadioCompanion
[13] Ettus, M. USRP user's and developer's guide. Ettus ResearchLLC.