Design of an SDR based FM transmitter

A Project Report

submitted by

KEVIN SELVA PRASANNA V

in partial fulfilment of requirements
for the award of the degree of

BACHELOR OF TECHNOLOGY

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS

June 2018

THESIS CERTIFICATE

This is to certify that the thesis titled **Design of an SDR based FM transmitter**, submitted by **Kevin Selva Prasanna V**, to the Indian Institute of Technology, Madras, for the award of the degree of **Bachelor of Technology**, is a bona fide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Prof.Devendra Jalihal
Project Guide
Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai

Date: June 2018

ACKNOWLEDGEMENTS

I would like to express my gratitude to Prof. Devendra Jallihal for providing the opportunity to work under his guidance on this project. His ideas and guidance have directed me throughout this project. This project has provided me the opportunity to learn about various aspects of hardware design and communication systems.

I would also like to thank Mrs.Sampoornam Srinivasan for her valuable guidance about PCB design and hardware testing. This project would not have succeeded without her help. I would also like to thank Mr.Sakthivel and others at my lab for their valuable guidance and help throughout the project.

I would also like to thank the various labs in the department including Space Lab, IE Lab, CEC who provided me with various hardware resources and guidance that helped me complete the project.

ABSTRACT

KEYWORDS: FM Transmitter ; Software Defined Radio; Amplifier Design; Hardware Implementation.

FM broadcasting is a broadcast media intended to broadcast high fidelity audio and textual data to audience over a geographical area. It employs Frequency Modulation technology to modulate audio and other data over radio waves in the VHF band. FM Receivers are simple, cheap, portable and battery operable. Hence in times of emergency and natural disaster, FM broadcast media could be a powerful tool in disseminating rescue information when other non-broadcast communication methods and the power grid has failed. A Software defined Radio is one in which standard radio communication techniques like modulation, filtering and detection which are conventionally implemented in hardware are instead implemented in software. This project involved the design, construction and testing of an SDR FM transmitter setup.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

FM is a simple and cheap technology used for broadcasting high fidelity audio over an area. It employs a technique called Frequency Modulation in which the frequency of the carrier wave is modulated according to the audio signal. Commercial FM standard allows for the broadcast of stereo audio upto a bandwidth of 15KHz in each channel. FM also allows for the broadcast of low bandwidth digital data carrying information like the station name, Alternative frequency and other data relating to the audio transmitted.

In times of emergency or a natural disaster, the power grid and mobile phone networks often fail. Effective quick dissemination of information is essential for rescue services in such situations. FM receivers are simple, cheap and battery operable. Moreover most mobile phones have FM receivers built into them. Hence FM technology would be the right candidate to broadcast essential information to people in a locality. Community Radios also use FM transmitters to communicate within a locality as the technology is cheap and simple. Such FM transmitters unlike commercial FM stations that blast out power in KiloWatts need just a few Watts of power to broadcast within a locality. A simple low cost FM transmitter could find use in the above situations.

A Software Defined Radio (SDR) performs filtering, mixing and other signal processing functions that are conventionally implemented in hardware, instead in software. Such radios are more flexible and configurable than conventional hardwired radios. This project makes use of an SDR to generate a frequency modulated RF signal. Since the power output of an SDR is not high enough for practical broadcast purposes, a VHF power amplifier was built to increase the power output to about 1-2 Watt. Two different setups were implemented, one using a USRP and one using a Raspberry Pi as an SDR. The output of the SDR was amplified by about 12dB and connected to an antenna. The obtained radiated power and range were observed.

1.1 The FM transmission Standard

Commercial FM radios employ frequency modulation technology to modulate a stereo audio signal onto a carrier wave. Broadcasting is carried out in the ITU Radio band 8 (VHF) from frequencies 88 MHz to 108 MHz.

In Frequency modulation, the instantaneous frequency of the carrier wave is modulated in accordance with the modulating audio signal. The transmitted RF signal can be expressed as

$$u(t) = A_c cos(2\pi f_c t + \theta(t)) \tag{1.1}$$

where f_c is the frequency of the carrier signal. The instantaneous frequency offset relative to the carrier is given by $\frac{1}{2\pi}\frac{d\theta(t)}{dt}$ (Madhow, 2014). Hence if $m(t)$ is the baseband modulating signal, then

$$\frac{1}{2\pi}\frac{d\theta(t)}{dt} = k_f m(t) \tag{1.2}$$

In India and in most other countries, k_f is chosen such that the peak frequency offset of the transmitted signal is 75 KHz. The baseband modulating signal is a multiplexed version of multiple signals including the stereo audio and the Radio Data System(RDS).

The stereo audio signal is used in the baseband modulating signal after being filtered by an pre-emphasis filter. The pre-emphasis filter is mathematically equivalent to a simple high pass RC filter of time constant $50\mu s$. This is performed to boost the energy in the high frequencies since in any FM system, noise is higher in the high frequency region.

For stereophonic transmission with Radio Data System (RDS), the baseband modulating signal is composed of 4 signals (fcc, 1983)

- A 15KHz band-limited audio signal which is the half of the sum of the left and right pre-emphasised audio channels.
- A pilot sine tone at 19KHz. This is to indicate the receiver that the FM signal is actually carrying a stereo (2-channel) audio signal. It will also be used by the receiver to actually demodulate the stereo audio signal.
- A 15KHz band-limited audio signal which is half of the difference between the

left and the right audio channels mixed onto a 38 KHz sub-carrier. The 38 KHz subcarrier is phase locked to the second harmonic of the 19KHz pilot signal

- RDS data stream mixed with a 57KHz carrier which is phase locked to the third harmonic of the 19KHz pilot signal.

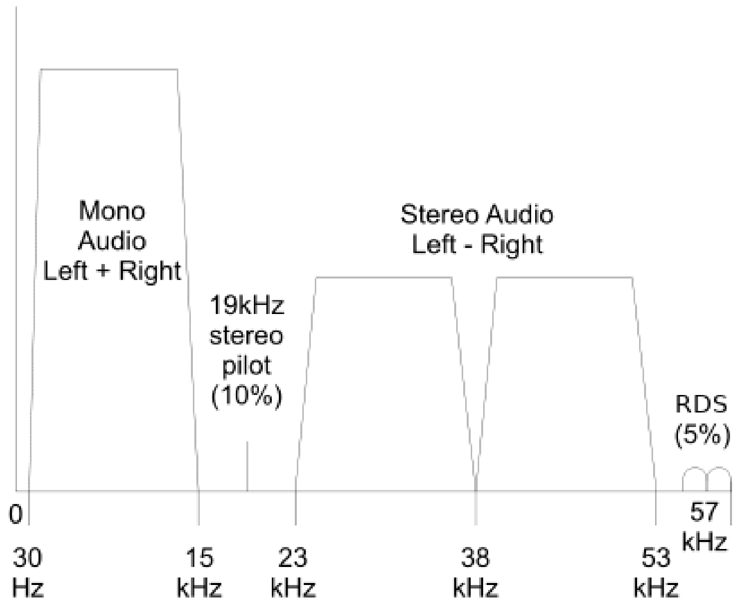

Figure 1.1: The various components of the modulating baseband signal

The Radio Data System is a standard for broadcasting low bandwidth digital information along with FM. It is amplitude modulated onto a 57KHz suppressed carrier using a shaped biphase coded (Differential Manchester encoding) digital data stream. Each bit is encoded onto 48 cycles of the sub-carrier implying a bit-rate of $57/48 = 1.1875kbps$ (rds, 1998). The RDS data is sent in groups of 104 bits. The Radio Data System supports various fields of data like Alternative frequencies, Clock time and date, Program type, Program Service Name, Radio Text and so on. Most FM receivers have only the ability to decode Program Service Name (PS Name) and Radio Text (RT) fields in RDS. Hence only those two fields were tested in the project although the software supports most other RDS fields.

CHAPTER 2

Viability study of RDS in Indian Languages

Broadcasting RDS data in Indian languages would be useful if the technology is used for rescue services in India. The RDS data fields of Program Service (PS) name and the Radio Text need to be broadcast in an Indian language to achieve this.

RDS data is sent in groups of 104 bits. Each group has 4 blocks of 26 bits each. The 26 bits is composed of 16 data bits and 10 parity bits. There are various types of groups and each type carries a field of RDS data. The RDS fields PS name and Radio Text are of prime significance as most FM RDS receivers including mobile phones decode only these two fields. The PS name is carried in group types 0A, 0B and the Radio Text field data is carried in group types 2A and 2B.

The Annexure E of the RDS specification document (rds, 1998) provides the repertoire of characters that are supported for transmission in the RDS standard. There are three code-tables of characters that include Latin, Cyrillic, Greek, Arabic and Hebrew characters. The code-table to be used for the character decoding is conveyed by the transmitter by the use of special repertoire control character. In the absence of a repertoire control character, the first code table is used as the default. The repertoire of characters does not include any of the characters from any Indian scripts. This is because the RDS standard was originally developed by the European Broadcasting Union (EBU) primarily for the European nations. Hence there is no direct way to transmit characters from Indian scripts through RDS.

But there are a few possible solutions that could be used to transmit characters from Indian scripts and decode them on FM receivers. A few possible solutions are

- RDS standard provides support for Open data applications that could be used for custom data broadcasts. This could be exploited to transmit Indian characters either in Unicode or in other encoding formats. But that would require modifying the receiver to be able to decode the open data application.
- If the receiver allows for changing its software, the Greek, Hebrew or other characters that RDS supports could be mapped and swapped with Indian language characters and displayed. But this too would involve changing the software on the receiver.

- RDS 2.0 standard, which was standardised in 2015 provides support for Unicode encoding. This would enable the use of all Indian language characters. But RDS 2.0 technology is still in its infancy and is yet to be implemented in receivers. Receivers could be modified to support RDS 2.0.

All solutions stated above require modifying atleast the receiver's software if not its hardware. But the principle form of FM receiver that is popular in India is a mobile phone FM receiver. The FM and RDS decoding on modern mobile phones happen either in hardware or in unmodifiable firmware. Fortunately, the second solution stated above would not require changing the decoding algorithm but just the application that displays the RDS data onto the screen. But modern phone operating systems like Android and iOS do not provide APIs (Application programming Interface) to access the RDS data and display it onto the screen (and, 2018). Most Android phone manufactures develop their own software stack and an application to access the FM decoding hardware and display the RDS data onto the screen. The software stack and APIs need to be reverse engineered in order for a custom application to be built that could access the FM hardware and this should be done separately for every phone manufacturer. There are a few projects and Android apps (nex, 2018) that have successfully accomplished this for many android phones but such projects are not open-source and hence cannot be customised.

Hence no practical solution to the problem of transmitting RDS data in Indian languages was identified during the project.

CHAPTER 3

FM transmission on a USRP

A USRP is a Software Defined Radio(SDR). In an SDR, signal processing and RF operations like modulation, filtering, decoding are performed in software and hence are highly customisable.

In this project, a USRP N210 and a USRP 2901 were used. Both these USRPs work together with a computer. The computer performs filtering and other operations and generates digital samples of the baseband signal and sends it to the USRP. The USRP has an FPGA and an RF frontend to mix the baseband signal with the carrier and generate the analog RF wave. The USRP N210 interfaces with the computer using a Ethernet connection while the USRP 2901 relies on a USB connection. The procedure to setup the connection between the USRP and a computer is outlined in Annexure A.

This project made use of an application software called GNU Radio to interface with the USRP. GNU Radio is an application that can perform SDR tasks like filtering, modulation and mathematical computations on a computer and send the generated digital samples to a USRP. GNU Radio has an easy-to-use interface and common signal processing operations are available as blocks that need to be dragged and dropped into the program. The application generates a python program that performs the operations on the computer and runs it.

FM transmission along with RDS as described by the standards was implemented in GNU Radio for the project. For the RDS data stream generation, an opensource GNU Radio library 'gr-rds' (gr-, 2018) was used. The GNU Radio block diagram for stereo FM transmission is shown in Figure 3.1. The 'gr-rds' library needs to be installed to use the block 'RDS Encoder' in the block diagram. The GNU Radio program generates baseband samples at a sampling rate of 1MSps and sends it to the USRP to perform an FM transmission at 107.9 MHz.

Anti-aliasing filters and Pre-emphasis filters have been neglected in the block diagram. This is because in the presence of those filters, the processing power of the

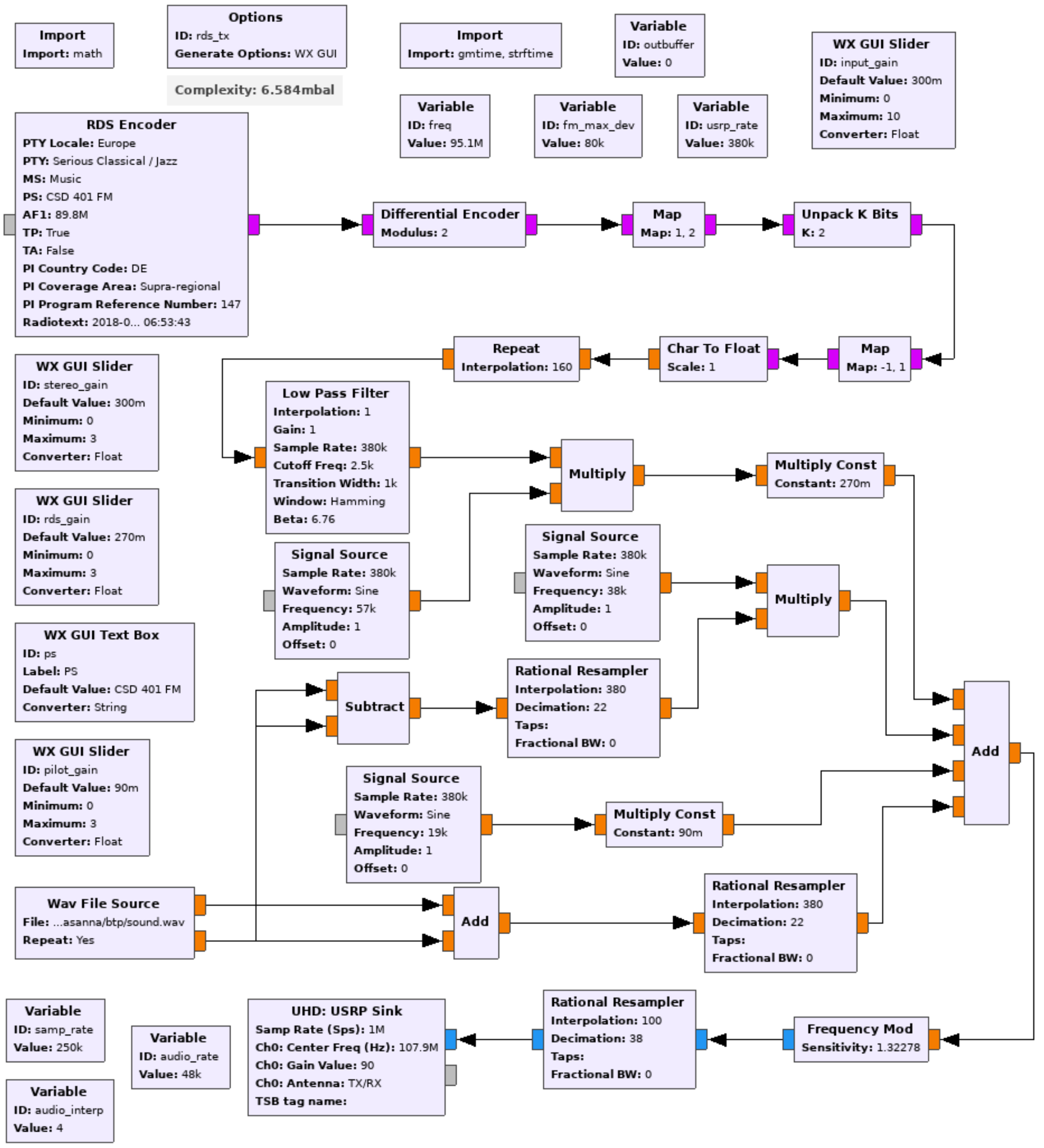

Figure 3.1: The GNU Radio block diagram for Stereo FM transmission with RDS

computer used in the project (Processor: i5-4210U) was not enough to compute those filters and generate samples at the required sampling rate. Those filters could be added to improve the quality of the audio. But the effect of neglecting the pre-emphasis and anti-aliasing filters were quite minimal and negligible.

The above GNU Radio program was tested on a computer running Linux with GNU Radio 3.7.11 and UHD 3.10.2 installed. The FM transmission was tested on an Android phone FM radio by connecting an antenna to the Tx connector of the USRP. The program was tested on a USRP N210 and on a USRP 2901. The procedure to setup the software and hardware to run the above program is provided in Annexure A.

When the output RF signal from the USRP 2901 was observed by a spectrum analyzer, an output power of 18 to 18.3 dBm was observed for different carrier frequencies in the FM band of 88 to 108 MHz. The output power obtained is clearly not sufficient for transmission over a long enough range and hence demands the need for a power amplifier.

CHAPTER 4

FM Transmission with a Raspberry Pi

The Raspberry Pi(Rpi) is a low cost microcontroller with an ARM processor and 40 GPIO pins. It can boot operating systems like Linux from an SD card. The 7th pin on Raspberry Pi's GPIO header is the clock pin used to generate a digital clock signal. The frequency of the clock signal can be controlled programmatically in software. Raspberry Pi has the ability to generate clock signals upto 100s of MHz and this feature could be exploited to generate a frequency modulated signal.

The opensource software project 'PiFmRds' (pif, 2018) is a demonstration of the above feature. The project supports Stereo FM transmission along with RDS. The 'PiFmRds' project's code was used in this project and it worked fairly well. When a long wire was connected to the pin 7 (GPIO 4) on the Raspberry Pi, FM transmission of stereo audio along with RDS data was achieved. The code is in C programming language and it was tested on a Raspberry Pi 2 with Raspbian Stretch OS installed.

As the Raspberry Pi's GPIO pins were not designed to provide high power, the RF power transmitted by it is very low. When the signal from the GPIO pin was observed with an Oscilloscope, a 1.2 V peak to peak signal was observed. Combined with the fact that the Rpi can provide a current of 16 mA on a single pin, about 5mW of RF power should be transmitted from the pin. This is too low a power for FM transmission. Hence a high gain power amplifier would be needed to amplify the signal for transmission.

Moreover, the Raspberry Pi generates digital square clock pulses on its GPIO pin. Hence the signal would be rich in harmonics. Hence the signal needs to be low pass filtered, to remove the harmonics. There is significant power especially in the odd harmonics.

CHAPTER 5

Design of a Power Amplifier

5.1 Schematic Design

A RF power amplifier is used to increase the power of weak RF signals. There are various classes of Power amplifiers depending on the bias point and the fraction of the time the transistor conducts. In this project, a simple single stage single transistor class B power amplifier was built.

In a class B power amplifier, the conduction angle is $180°$ i.e., the transistor conducts for one half of a wave cycle. As the transistor is off for the other half of the wave, the efficiency is improved. But this comes at the cost of quality as the output wave will have only half of the input signal. The other half is reconstructed using a resonant LC oscillator. As the signal is frequency modulated, linear operation of the transistor is not crucial unlike a signal that is amplitude modulated. The signal is then filtered and passed through an LC impedance matching network. The input impedance of the amplifier also needs to be matched using an LC impedance matching network to prevent reflections at the input. A choke coil was connected on the way to the power source to prevent the RF signal from reaching the power source. The circuit is designed to operate at a supply voltage of 12 V.

The MOSFET RD06HVF1 was chosen as it was an RF power amplifier capable of providing upto 10 W of power. The circuit was designed and simulated in Spice using the software LTSpice. The values were initially arrived by rough hand calculation and later tuned and simulated in Spice to achieve the best possible results. They were later adjusted to the values of components available in the market. The schematic of the designed Class B power amplifier is shown in figure 5.1.

In the figure 5.1, the capacitor C8 is a decoupling capacitor, L3 is a choke coil to prevent RF signal from reaching the power supply, C5 is a decoupling capacitor to short out the RF signal to ground.The resistor R3 serves to reduce the transients when

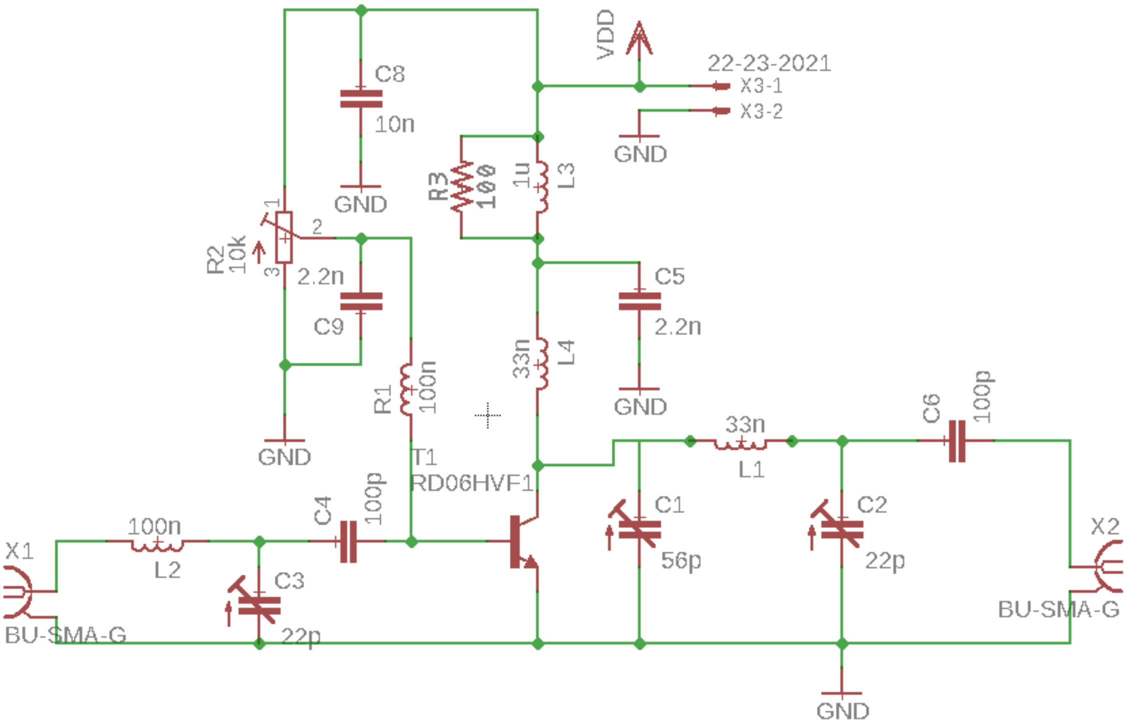

Figure 5.1: Schematic of the power amplifier

turning on the input signal. R2 is a potentiometer used to set the bias voltage. C9 is a decoupling capacitor to decouple the bias point from RF. R1 is an inductor that chokes the RF from reaching the biasing network while setting the bias voltage on the gate of the MOSFET. C4 is a input decoupling capacitor to block DC. C3 and L2 form the input impedance matching LC network. L4 and C1 form a resonant LC circuit resonant at close to 100 MHz. L1 and C2 match the output impedance and also act as a low pass filter at the same time filtering out the higher order harmonics. C6 is a decoupling capacitor blocking out DC from the output.

Simulations in Spice showed upto 14 dB of gain between the input and output if the gate bias voltage was set to about 3.6 V. This bias voltage was chosen to operate the device in Class B mode. The input and output voltage signal simulated in LTSpice is shown in Figure 5.2.

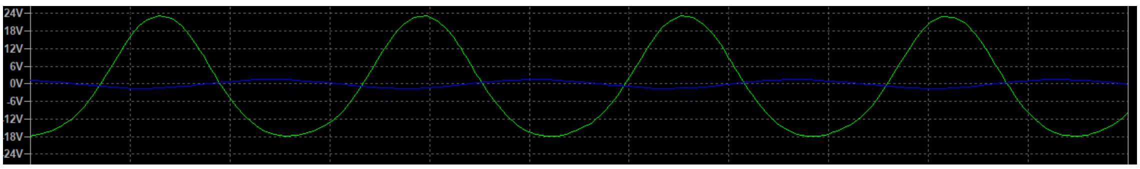

Figure 5.2: Schematic of the power amplifier

5.2 PCB Design

The power amplifier schematic shown in Figure 5.1 was made into a Printed Circuit Board (PCB). In any RF circuit, the PCB layout is important as interference and parasitic impedances introduced in the PCB could have a drastic impact on such circuits. Hence the PCB was designed according to RF PCB guidelines.

SMD components were chosen as much as possible to avoid parasite impedances introduced by through-hole components. The traces were kept as short as possible to prevent parasitic impedances and transmission line effects. Impedance control was not performed on the traces as the length of the traces were in the order of $\lambda/100$ and hence transmission line effects could be neglected. Moreover, impedance control would require a very thin PCB and thick traces for a 2 layer PCB. The width of the traces were designed to be 0.8 mm for lines carrying the input signal and 1.3 mm for the power lines and the lines carrying the amplified output signal. Sufficient clearances were provides between the traces. The second layer of the 2 layer PCB was entirely a ground plane. The ground plane helps in preventing EMI and prevents interference between traces. A two pin connector was chosen to connect the power supply. Right angled SMA connectors were chosen to provide the RF input and take out the output. Space was provided in the PCB for mounting a 1 inch wide heat-sink. Two mounting holes were made for the heat-sink and two mounting holes for the PCB itself.

The dimension of the designed PCB are 50.0 mm x 33.0 mm. The thickness of the PCB was chosen to be 1.6 mm. Some SMD components were not available easily in the market and hence through-hole alternatives were used. The components were soldered and connections were checked. Two PCBs were made with slight modifications in the component values. The second PCB designed had L4=60 nH and L1=70 nH with everything else remaining the same. This PCB will be henceforth referred as PCB 2. The design of the PCB is shown in Figure 5.3. A picture of PCB 2 is shown in Figure 5.4.

There was an error in the design of the PCB layout. The pins for the transistor were arranged wrong i.e, the pins for the drain and source were interchanged in the PCB layout. To remedy this, the pins of the transistor were twisted and crossed before soldering to the PCB. This seemed to work well and give good results.

At a bias voltage of 3.6V, a gain of about 12 dB was observed from the designed power amplifier when the input power was close to 60 mW. The gain of the amplifier changes with the bias voltage as well as the input signal. If the bias voltage is increased, a gain of about 14 dB can be achieved at the cost of efficiency provided the heat sink can handle the excessive heat dissipation. As the transistor is not exactly linear, the gain also varies with the power of the input signal. More detailed results are provided in Chapter 6. The output impedance is matched to close to 50 ohms. The input impedance matching network was neglected in a few tests as the its effect was minimal.

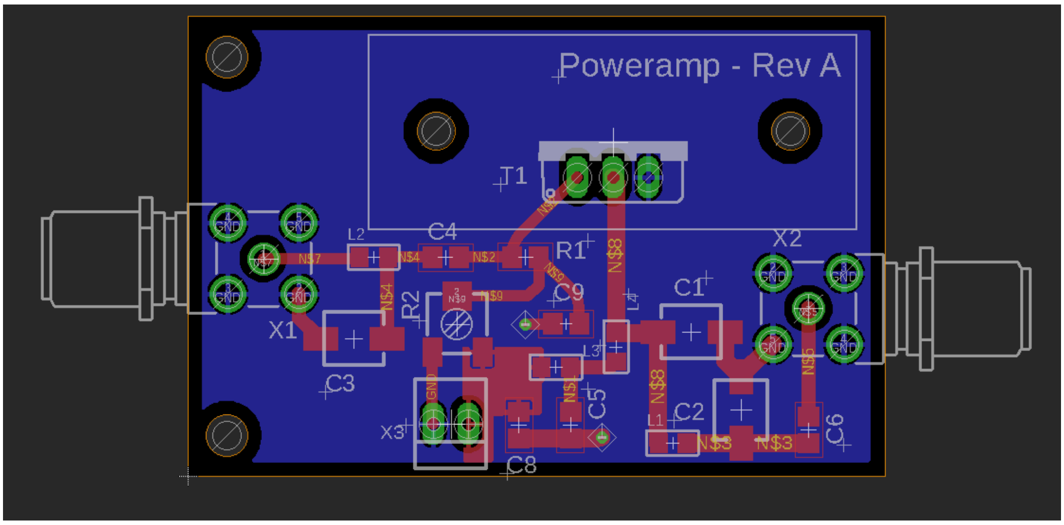

Figure 5.3: PCB Layout of the power amplifier

Figure 5.4: The power amplifier (PCB 2)

CHAPTER 6

Tests and Experiments

Two setups were made, one with a USRP and one with a Raspberry Pi connected to the power amplifier. Various tests and experiments were performed to measure the output power, spectrum and range of transmission. The setup and the results of those tests are detailed in this chapter.

6.1 The USRP setup

As the power output from the USRP is about 60mW at 100 MHz at maximum gain setting, a power amplifier is required to boost the output power. Hence the output of the USRP can be connected to the power amplifier input using an SMA cable and the output of the power amplifier could be connected to an antenna using another SMA cable. The setup is as shown in Figure 6.1.

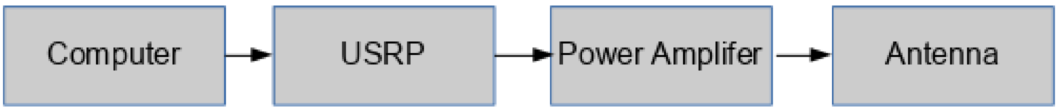

Figure 6.1: The setup with USRP and power amplifier

When the power amplifier was connected to the USRP, an output power of about 1.3 W was observed at the output of the power amplifier with a load of 50 ohms. The second PCB (PCB 2) with different inductance values was not perfect and the output power was close to 800 mW. Moreover a power of about 1-2% was observed in the second and third harmonics. Hence a THD of about 1 to 2% has been observed. The picture of the setup is shown in Figure 6.2.

The output of the power amplifier was connected to a spectrum analyzer and the output power and spectrum were analyzed. The output power obtained for different cases is tabulated in Table 6.1 and Table 6.2 for the two PCBs designed.

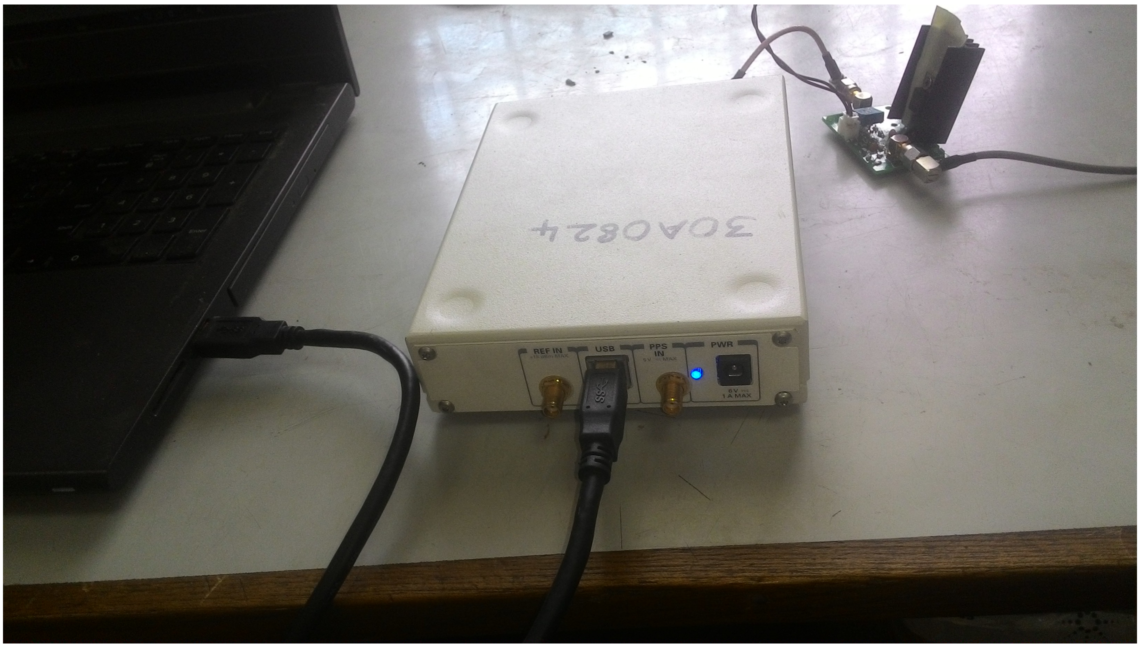

Figure 6.2: A picture of the setup with USRP 2901 and power amplifier (PCB 1)

Table 6.1: The output power from the power amplifier (PCB 1) for different power input at different carrier frequencies from the USRP 2901

Input power (dBm)	Operating Frequency (MHz)	Output power (dBm)
18.3	95.1	31.47
18.0	107.9	30.63

Table 6.2: The output power from the power amplifier (PCB 2) for different power input at different carrier frequencies from the USRP 2901

Input power (dBm)	Operating Frequency (MHz)	Output power (dBm)
18.3	95.1	28.75
18.0	107.9	28.12

The efficiency of the amplifier at a bias voltage of 3.6 V was about 36%. At 12 V, the amplifier drew 0.30 A of current and produced 1.3 W of RF power (PCB 1) when excited by an input signal from the USRP. This provides an efficiency of 36%. The remaining 2.3 W is dissipated as heat and the heat sink needs to be able to handle this. The transistor can tolerate a maximum temperature of 150° and hence the thermal conductivity of the heat sink needs to be ensured to maintain a low enough temperature.

The Figure 6.3 shows the spectrum around 95.1 MHz for the output of the power amplifier made from PCB 1. Note that each horizontal division in the graph corresponds to 1 MHz and a 40 dB attenuator was used to measure the spectrum.

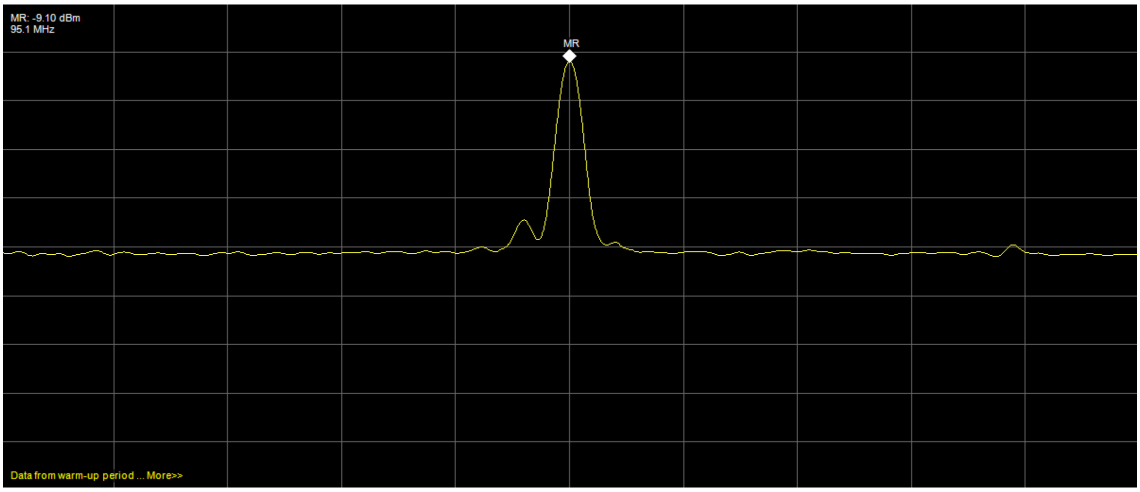

Figure 6.3: The spectrum from 90 to 100 MHz of the output signal from PCB 1 power amplifier

The Figure 6.4 provides the extended spectrum upto 350 MHz showing the second and third harmonics. Note that each horizontal division in the graph corresponds to 50 MHz. Also note the power in the second and third harmonics.

The range of transmission was also tested using an Android phone. With the USRP and an antenna connected to the power amplifier, the range of transmission was tested. Different types of antenna including a folded dipole antenna, random wire monopole antenna were tested. A monopole antenna of about 75 cm length constructed using a random wire gave the best results for range.

VHF transmission usually requires installing an antenna at a considerable height above the ground to maximize Line of Sight propagation. This is because VHF transmission usually do bend a lot and are absorbed by concrete walls. But tests in the

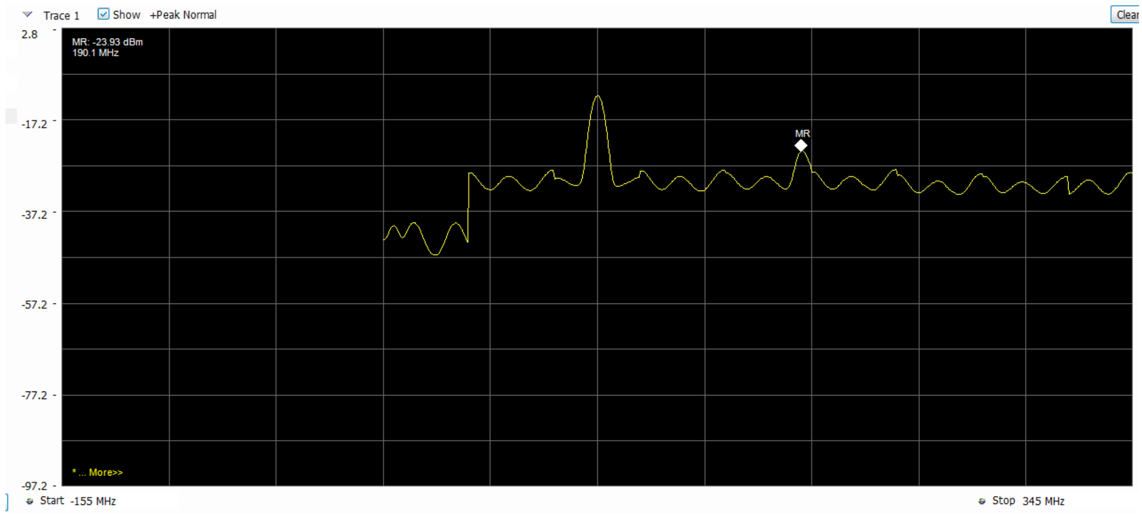

Figure 6.4: The spectrum from 0 to 350 MHz of the output signal from PCB 1 power amplifier

project were mostly done inside a lab (CSD 401) and range tests were carried out in and around the Electrical Sciences Block (ESB). The signal was not able to cross many walls and indoor reception of signal was quite not possible. But the signal was received in multiple places at 300 to 500 m away from ESB where Line of Sight was available. The signal was received at places including the Football ground and Bhadra Hostel which are about 300 to 500 m from the transmitter antenna. If the transmitter antenna was installed at a high enough location, reception is possible at further distances with a transmitted power of 1.5 or 2 W.

For emergency services, a range of a km may not be enough at all times. A range of atleast 5 to 10 km may be desirable in such situations. In such cases, transmitted power of more than 10 W would be required. This would required a high gain or a multiple stage amplifier. If the intended reception is to happen only in a single direction from the transmitter, a high gain directional antenna could be used.

6.2 The Raspberry Pi setup

The Raspberry Pi being a fully functional computer all by itself can replace the computer and the USRP in the previous setup. Raspberry Pi was not designed to provide RF output. Hence taking out the RF signal from the Raspberry Pi is not easy. In this project, the signal from the pin 7 (GPIO 4) of the Raspberry Pi was directly connected to the

SMA connector on the power amplifier using a short wire. The wire itself can provide radiation resistance and radiate RF power and hence was kept as short as possible. A common ground was also ensured between the Raspberry Pi and the Power amplifier. The setup is as shown in Figure 6.5.

Figure 6.5: The Raspberry Pi setup

As the power output from the Raspberry Pi was small, the final amplified output from Raspberry Pi is also small. The power amplifier was initially designed only for the USRP and hence the gain of the amplifier is also less with the Raspberry Pi. The power of the output signal from the Power Amplifier measured using a spectrum analyzer is tabulated in Table 6.3.

Table 6.3: The output power from the power amplifier (PCB 1) for different power input at different carrier frequencies from the Raspberry Pi

Input power (dBm)	Operating Frequency (MHz)	Output power (dBm)
7 (estimated)	95.1	17.81
7 (estimated)	107.9	17.63

The range of transmission was also considerably less for the Raspberry Pi. At places with Line of Sight, a range of about 50 to 100m was observed at various locations in and around the Electrical Science Block.

The Power amplifier has a low pass response that was meant to suppress the higher order harmonics. Since the RF signal from Raspberry Pi has considerable power in the higher harmonics, the power amplifier removes it. The spectrum of the signal obtained from the Raspberry Pi as recorded by a spectrum analyzer is shown in Figure 6.6.

A picture of the Raspberry Pi setup is shown in Figure 6.7.

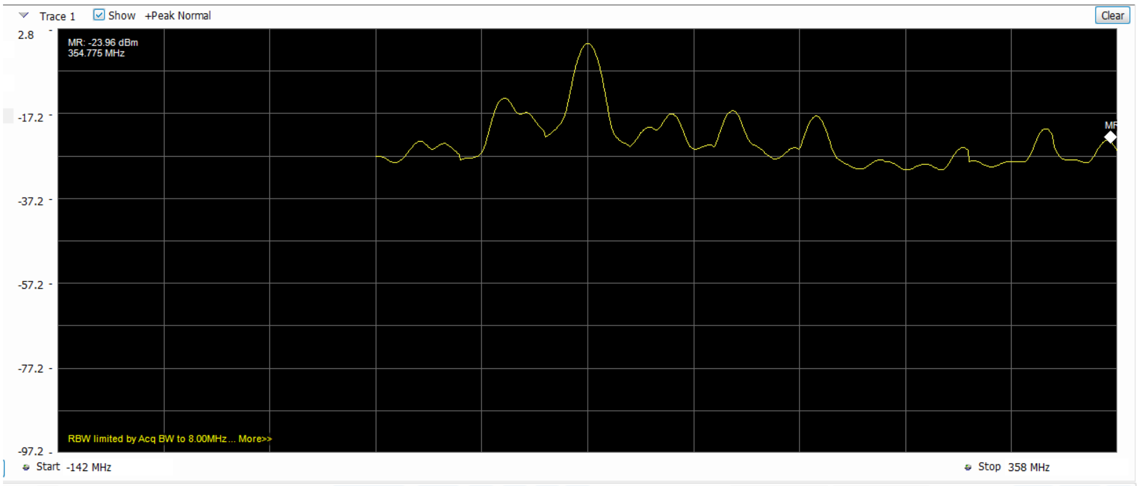

Figure 6.6: Spectrum of the signal from Raspberry Pi from 0 to 350 MHz (Carrier Frequency = 107.9 MHz)

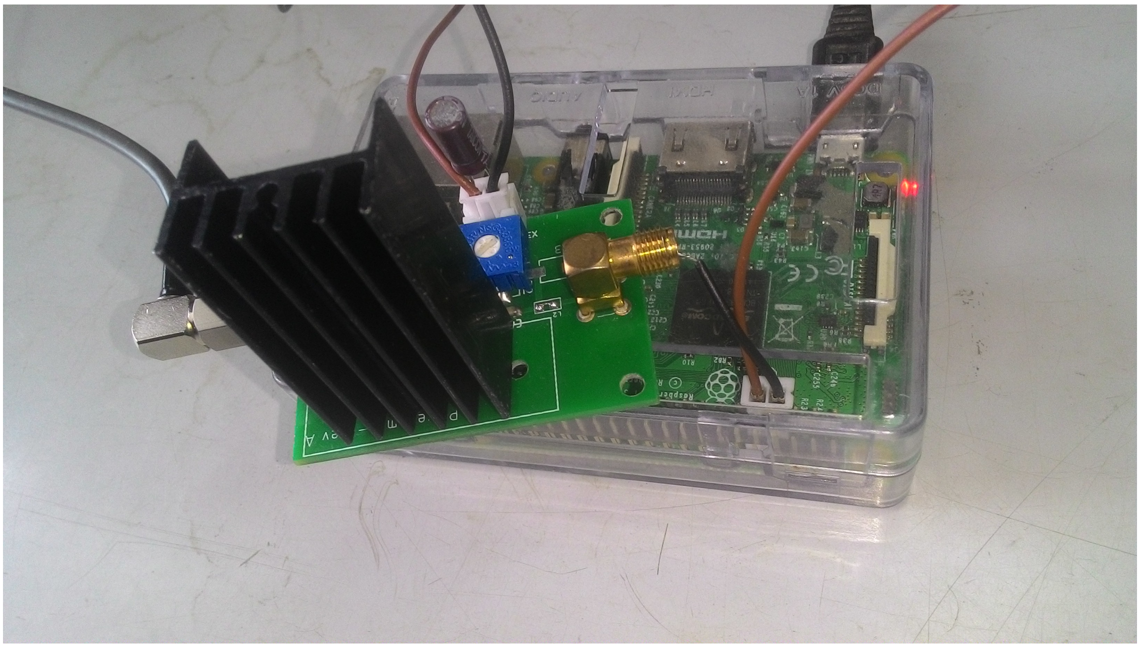

Figure 6.7: A picture of a Raspberry Pi 3 and the power amplifier working

CHAPTER 7

Conclusion

The USRP setup provides a power of 1.3 W when the bias voltage is 3.6 V. By using a higher bias voltage and a bigger heat sink, the transmitted RF power could be increased to upto 2 W or more. This could provide a range of 1 to 2 km if the transmitter antenna is installed at a high enough place. But if a range of about 5 km or more is desired for rescue services in emergency situations, power output of 10 W or more is needed and hence an amplifier of higher gain would be needed.

The Raspberry Pi setup is a lot cheaper and simpler than the USRP setup. It takes off the cost of the USRP and the computer and is a lot more portable. But the power output of the Raspberry Pi is an order of magnitude less than that of the USRP and hence the range is also less. A more powerful amplifier is required to amplify the signal to a reasonable level.

7.1 Issues and Limitations

- The output power of the setup may not enough to cover a wide enough geographical area for rescue services.
- Better impedance matching is required to increase the transmitted power as the there is a slight impedance mismatch at the output of the amplifier in the current setup. This is because the datasheet of the MOSFET RD06HVF1 does not provide a lot of information required to perform impedance matching.
- Inspite of the large choke coil and the decoupling capacitor, RF signal somehow finds its way into the power supply. The exact reason for this is not entirely known. This could cause problems with some power supply.

7.2 Implications

Low power FM transmitters could be used with for disseminating rescue information at times of emergency. FM operates at a much lower frequency than other services like

GSM or LTE and hence covers a wider geographical area. Moreover the simplicity and ubiquity of FM receivers also make it a good choice.

Low Power FM Radio Stations (LPFMRS) are low power radio stations operated by communities, educational institutions and Farm Radio broadcasters. Indian Government provides license for LPFMRDS to transmit power upto a power of 50 W(lpf, 2004). Cheap low power FM transmitters could be used by such community radios in place of costly equipment costing lakhs of rupees.

Digital Radio Mondiale (DRM) is a digital Radio standard which is poised to succeed analog FM transmission. DRM has better bandwidth efficiency and other allied services. All India Radio (AIR) has the largest DRM network in the world(drm, 2014). Since this project uses software (SDR) for modulation, changing the setup to DRM Radio would just involve a software upgrade.

The project could also be used as an FM repeater to be used in places where FM reception is not available or is poor. As online streams of FM stations, podcasts and TV audio streams are available, Raspberry Pi or a computer could be programmed to fetch these online streams and broadcast them using the FM transmitter setup.

7.3 Possible Future work

RDS 2.0, announced in 2015, provides support for UTF-8 encoding. This would provide support for Indian characters. It also has a higher data rate and better emergency features. But the technology is pretty new and closed and is not certain if it will replace RDS 1.0. This project could be upgraded to support RDS 2.0 in the future with a software upgrade. RDS 1.0 also supports an emergency feature called Emergency Warning System (EWS) for compliant receivers. But the feature is not supported on most receivers.

DRM and other digital radio standards could be implemented on software with GNU Radio or on the Raspberry Pi. The hardware setup is completely compatible with those standards.

FM transmission took less than 50% of the CPU processing power on the Raspberry Pi. This means that cheaper micro-controllers with less powerful hardware could be used to replace the Raspberry Pi. Orange Pi could be a likely candidate(ora, 2018).

A pre-amplifier could be connected with the Power amplifier when a Raspberry Pi is used so that a higher gain can be accomplished.

APPENDIX A

Instructions to connect USRP to computer

This project was tested on a USRP 2901 and on a USRP N210. The USRP N210 interfaces with the computer through an Ethernet interface while the 2901 uses a USB cable. This project was tested on a computer running Arch Linux and on a computer running Ubuntu 18.04. Before using either of the USRPs, USRP Hardware driver(UHD) and GNU Radio need to be installed.

To use the USRP N210, the computer's IP address must be manually set to 192.168.10.1. On a Linux computer, this can be done by using the command

```
sudo ip addr add 192.168.10.1 dev eth0
```

After this, the connection with the USRP can be checked with the command

```
uhd_find_devices
```

USRP 2901 uses a USB connection to interface and hence requires no additional configuration. The connection can be checked with the same command as shown above. Note that every time, USRP 2901 is switched on, an image needs to be flashed onto it which UHD automatically takes care of. But note that this can take some time. Also note that UHD identifies USRP 2901 as a USRP B210 which is Ettus's equivalent of NI's USRP 2901.

The GNU Radio program provided in Chapter 3 takes in audio through a wav file. Note that the audio can also be provided through a microphone connected to the computer by replacing the 'Wav File Source' block with an 'Audio Source' block.

Also note that if the antenna connected to the power amplifier is brought too close to the USRP or the USB cable, the USRP can shutdown due to EMI. Hence proper distance has to be maintained between the antenna and the USRP cable.

APPENDIX B

Instructions to use the Raspberry Pi

This project has been tested on a Raspberry Pi 2 and on a Raspberry Pi 3 running Raspbian Strech OS. However the code is compatible with any Raspberry Pi running any variant of Linux. An image of the operating system needs to downloaded and flashed on the SD card. This can done by using the command

```
dd bs=4M if=2018-04-18-raspbian-stretch.img of=/dev/sdX
                                        conv=fsync
```

where '2018-04-18-raspbian-stretch.img' should be replaced with path to the image file and '/dev/sdX' replaced by the SD card's device file.

In this project, the Raspberry Pi was programmed by using another computer through an Ethernet connection. The Raspberry Pi could be connected to a HDMI monitor and a keyboard and then programmed for convenience.

To setup the 'PiFmRds' code on the Raspberry Pi, run the following commands after logging into a Raspberry Pi running Raspbian

```
sudo apt-get install libsndfile1-dev
git clone https://github.com/kevinselvaprasanna/PiFmRds.git
cd PiFmRds/src
make clean
make
sudo ./pi_fm_rds -audio sound.wav -freq 107.9 -ps [PS_NAME]
                                        -rt [RADIO_TEXT]
```

To make the Raspberry Pi automatically start the FM transmission on booting, add the last command 'sudo ...' with an '&' sign at the end of the line to the file '/etc/rc.local' on the Raspberry Pi.

By default, Raspberry Pi would broadcast the audio file that is provided in the above command after the '-audio' parameter. Raspberry Pi cannot take in analog input. But a digital USB microphone or a USB sound card with an analog microphone could be connected to the Raspberry Pi to transmit the audio from the microphone on FM. To do so, an USB sound card with a microphone needs to be plugged into the Raspberry Pi and the following command needs to be run

```
sudo arecord -fS16_LE -r 44100 -Dplughw:1,0 -c 2 -  | sudo
./pi_fm_rds -audio -
```

It is also possible to broadcast and download a live audio stream from the internet and transmit it. All these were tested during the course of the project.

REFERENCES

1. (1983). *1983 rules for subcarriers*. Federal Communications Commission. URL https://www.fcc.gov/sites/default/files/rules1983subcarrier.pdf.

2. (1998). *Specification of the Radio Data System (RDS) for VHF/FM sound broadcasting in the frequency range from 87,5 to 108,0 MHz*. European Broadcasting Union. URL http://www.interactive-radio-system.com/docs/EN50067_RDS_Standard.pdf.

3. (2004). Low power fm radio:indian universities jump into broadcasting. URL http://egov.eletsonline.com/2004/08/low-power-fm-radioindian-universities-jump-into-broadcasting/.

4. (2014). Drm india page. URL http://www.drm.org/drm-india-page/.

5. (2018). *Android Developer Guides*. Google Developers. URL https://developer.android.com/guide/.

6. (2018). gr-rds github repository. URL https://github.com/kevinselvaprasanna/gr-rds.

7. (2018). nextradio website. URL http://nextradioapp.com/.

8. (2018). Orange pi website. URL http://www.orangepi.org/.

9. (2018). Pifmrds github repository. URL https://github.com/kevinselvaprasanna/PiFmRds.

10. **Madhow, U.**, *Introduction to Communication Systems*. Cambridge University Press, 2014.