DESIGN OF AN INTRANET RADIO USING REAL TIME TRANSPORT PROTOCOL FOR UNIVERSITY CAMPUS

by

Nsindiso Mnqobi Matshanga R146942M

Submitted in partial fulfillment of the requirement for the degree of

Bachelor of Science (Honours) Telecommunications

Department of Applied Physics & Telecommunications in the Faculty of Science and Technology at the

Midlands State University

GWERU

June, 2018

Supervisor: Dr Action Nechibvute

Co-supervisor: Mr. Bernard Siachingoma

Abstract

The dissipation of information is an essential aspect of any community. The advent of media services such as television and radio has improved the way information has been broadcasted to the general public. Unfortunately in Zimbabwe Universities, these services have not yet been utilized. This Dissertation illustrates a prototype of campus radio streaming system over an intranet. This is to provide local universities with a cheaper alternative to providing a local radio station compared to the traditional terrestrial radio. The project demonstrates a system using application layer routing for real time audio streaming which takes advantage of the already existing network infrastructure. This system could be used to build a local radio station for universities.

To Ntugamili and Saneliso Matshanga.

DECLARATION

I, Nsindiso Mnqobi Matshanga, hereby declare that I am the sole author of this thesis. I authorize the Midlands State University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Signature ___________________________________ Date ______________________________

APPROVAL

This dissertation/thesis entitles “Design of an Intranet Radio for University Campus” by Nsindiso Mnqobi Matshanga meets the regulations governing the award of the degree of B.Sc. of Telecommunication’s honours degree of the Midlands State University, and is approved for its Contribution to knowledge and literal presentation.

Supervisor …………………………………………………………….

Date…………………………………………………………

ACKNOWLEDGEMENTS

I would like to give my sincere gratitude to all the people that have influenced me directly and indirectly to get to this level in my academic endeavors. My greatest gratitude and Love will and always go to GOD for blessing me with numerous gifts that I am truly and eternally grateful for.

I thank my family for all their love, support and encouragement they have given me throughout this whole process. My Friend’s and fellow classmate’s whose assistance was invaluable.

Lastly and most importantly I would like to thank Dr Action Nechibvute, Mr. Bernard Siachingoma and all the department lecture for their support and assistant’s in the development and completion of this dissertation.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF GRAPHS

CHAPTER 1

1 INTRODUCTION

1.1 Background

University can often be a stressful experience for students and radio can provide a real outlet for creativity and can help break-up the stresses of studying and worrying about handing assignments in on time [1]. Radio stations can also act as great way to dissipate news around the campus. An hour of dedicated to university news is a good way to keep students and the wider community informed of developments at the university [2]. Radio station also gives a platform for those looking on breaking in to the entertainment industry an opportunity to express and develop their talents. [3] Stating it can be used to establish a creative hub within your university allowing students to express themselves and build their confidence.

Zimbabwe has 15 licensed FM radio stations and 8 licensed internet radio stations [4] and 8 community radio station's according to [5] but unfortunately [6] states that there are no bona fide licensed community radio station's in the country this being largely due to the tight broadcasting laws. This is also true for campus radio stations. According to [7] there are over 15 university campuses in Zimbabwe, with none of them operating a radio station.

According to [8] "IBM Community Radio is the new intranet radio station for IBMers worldwide, broadcast from Austin, Texas. The brainchild of a designer with the relatively new IBM Design group, up to 6000 employees tune-in to IBM Radio every day, and can talk about whatever they want, from 9-6pm CST." With the successful implementation of intranet radio at IBM, university campuses can utilize their intranet's to establish a campus radio station.

1.2 Overview of existing radio systems

FM radio [8] was the first technology used to broadcast audio. It is used to provide high quality sound over broadcast radio. The audio file is converted from analogue to digital, the analogue file is then modulated using frequency modulation. This radio's usually use VHF frequencies.

The disadvantage with building your own FM station is the huge cost and the legal issues in transmitting FM signals.

[9] Describes internet radio as a digital audio service broadcast transmitted via the internet. Internet. The user sends a request to the server through HTTP request and audio packets are sent through HTTP response to user. The problem with this is that when the internet is down there audio will not be received.

Intranet Radio is a digital audio service broadcasted and transmitted via an organisation private network. The intranet connects the client to the server. The client will send a request to the server through a HTTP or RSTP request and audio packets are sent though the HTTP or RSTP response to the client. This has the advantage that outside audience cannot access the content and when the internet is down audio will still be received.

1.3 Aim of study

Develop a model of a campus radio station which will utilize existing infrastructure to broadcast content.

1.4 Purpose of study

To provide the university campuses with a radio platform which will give campus community local entertainment, news and an opportunity for students trying to establish themselves in the entertainment industry a platform to share their talents while utilizing existing campus network infrastructure.

1.5 Objectives

- Design and build a Streaming server using a raspberry pi which would be acting as a dedicated computer with encoder and the streaming server.
- The server (Raspberry Pi) should send the audio data stream via an Ethernet cable to the wireless router.

- The router should then route the stream demonstrating wired and wireless communication. The stream will be sent via an Ethernet cable to an internet radio which would be built by a Raspberry pi, LCD Screen and speaker.
- Transmitting the stream via unlicensed wireless frequency bands 2.4 GHz which is declared by [10] that the frequency bands 2400 - 2483.5 MHz (ISM), 5.15 – 5.35 GHz and 5.470 –5.725 GHz are designated for the use of Short Range Devices (SRDs) including Radio Local Area Networks (RLANs).
- An Android Application will be used which will act as the player on the listener's smartphone.
- This Application will translate the audio data stream from the server and translates it into the sound heard by the listener.

1.6 Hypothesis

The project proposed will demonstrate a system which utilizes a university local network to broadcast radio content to students and lecturers.

1.7 Problem Statement

In Zimbabwe no campus currently has a local radio station this stations are seen as training ground for students looking on establishing careers in the broadcasting industry thus there is limited of no platform for students to share their talents. Other major disadvantage's lack of entertainment and there is no way of dissipating news to large number of audience around campus. With the use of optical fiber at university campuses, the bandwidth of networks has increased with a large chunk of the bandwidth not being fully utilized thus adding extra voice traffic will help utilize the bandwidth.

1.8 Justification

There is a need for a radio station around campus which will provide entertainment, local news also utilizing the available network infrastructure.

1.9 Dissertation overview

Chapter one – Introduction

Chapter two – Literature review

Chapter three – Methodology

Chapter four – Results and Analysis

Chapter five - Conclusion

REFERENCES

[1] David Wilkinson . "Online College Radio Benefits You Need to Know. Internet: https://radio.co/blog/online-college-radio-benefits-know/, April. 25, 2016 [April, 4 2018].

[2] Benjamin Miller. "Untapped potential of campus radio. Internet: https://www.universityaffairs.ca/opinion/in-my-opinion/untapped-potential-campus-radio/ Feb.18, 2015 [April, 4 2018].

[3]N.A. "List of Radio Stations in Zimbabwe. Internet: https://en.wikipedia.org/wiki/List_of_radio_stations_in_Zimbabwe./March.3, 2018[April, 4 2018].

[4] Maston Mbewe . "Community Radio Stations Issued Licenses By BAZ

. Internet: http://www.creativeloop.co.zw/2015/03/8-community-radio-stations-issued-licenses-baz/March.3, 2015 [April, 4 2018].

[5] Dakarai Mashava. "Controversy over community radio stations

. Internet: https://www.dailynews.co.zw/articles/2017/09/30/controversy-over-community-radio-stations Sept.30, 2017 [April, 4 2018].

[6] John Masuku. "ZACRAS-Zimbabwe's community radio stations in waiting," n.d, pp.1.

[7] MUNESU NYAKUDYA. "Community radio initiatives to start operating

. Internet: https://www.newsday.co.zw/2017/12/community-radio-initiatives-start-operating/

Dec.02, 2017 [April, 4 2018].

[8] Ahmad Rahmati1, Lin Zhong. Enabling Pervasive Mobile Applications with the FM Radio Broadcast Data System. IEEE, 2010.

[9] N.A. "Internet Radio: https://en.wikipedia.org/wiki/Internet_radio [April, 5 2018].

[10] Postal and Telecommunications Regulatory Authority of Zimbabwe (POTRAZ), Public notice, Harare, 2011.

CHAPTER 2

2 LITERATURE REVIEW

2.1 Benefits of campus radio

The first ever College Radio was called 9XM which was aired at the University of Wisconsin in 1912 [1]. Since its inception college radio has been seen as a tool which has been used to enhance the learning experience of students. [2] Stated that establishing college radio as an instructional medium it can change the learning experience, trigger active participation, enhance educational experience, and maximize community engagement.

Advantages of college radio includes advising the learning of differing language's, sharing of information between students and lecture's, broadcasting campus news and upcoming events, influencing students mindsets on issues that affect their overall performance in education and job hunting and providing a voice and a sense of ownership for students through participation.

With the cultural diversity in Zimbabwe which has more than 16 being spoken languages [3] e.g. Shona, Venda, Nambya, Chewa and Ndebele to name but a few. University is a unique community in which individuals from differing backgrounds meet with the goal of obtaining a converted degree. Campus radio could be a great tool for individuals to learn differing languages thereby enhancing the educational experience. By exploring the role of radio in imparting education at Allama Iqbal Open University in Pakistan and Al-Qudus Open University Jordan, emphasizing the importance of radio and television in "promoting educational process, improving curriculum and methodology" for students[4].

IRI (Interactive radio instruction) is a tool used that combines radio broadcasting and active learning with [5] saying it is among the most widely used tools in the develop world due to its good measure of funding by the united states Agency for international development. This tool could be used to offer content to students that could not be obtained during lectures such as computer literacy for students not doing a computer related program. There was convincing evidence of IRI’s positive impact on improvement of student learning outcomes on varying levels and extents [6].

The role of community media in political and social development in Jordan, highlighted the role of school radio in giving students the platform to share “badly needed” local cultural programs that exposed new artists and literary talents to the students” in addition to sharing local sports and news coverage [7]. University in Zimbabwe offer a wide range of program’s as college radio station’s offers student’s in varies programs a training ground to practice their profession’s e.g. at Midlands State university students doing media studies can run the campus news, Students doing Musicology can share their music and students doing telecommunications and computer science can maintain the equipment.

Participation in running the college radio not only develops a higher quality of professionals, but influences the mindsets of students by influencing student attitudes towards more important and rewarding issues thus empowering students. Held that participatory communication is an approach capable of facilitating people's involvement in decision-making about issues impacting their lives - a process capable of addressing specific needs and priorities relevant to people and at the same time assisting in their empowerment [8].

2.2 Contemporary Radio Transmission

2.2.1 Terrestrial Broadcasting

This type of radio broadcasting can either be analogue or digital:

- Analogue terrestrial radio broadcasting consists of a transmitter and receiver. Transmission sends the sine wave and is modulated via Analogue/Frequency Modulation. Analogue modulation has a frequency range 49MHz and Frequency modulation 800MHZ.
- Digital Terrestrial radio compresses the audio and transmits it from an antenna in pieces. The receiver then captures these pieces like it would on an analog signal but instead of playing them, it decodes the encoded audio.

Table 2.1: Difference Between Analogue And Digital Terrestrial Radio

Analogue terrestrial radio	Digital terrestrial radio
High interference	Low interference
High use of bandwidth	Low use of bandwidth
Does no allow compression	Allows compression
Does not allow Encryption	Allows Encryption
Variable Signal	Discrete Signal

Advantages of digital terrestrial radio over analogue terrestrial Radio

- *High Sound quality.*
- *More channels.*
- *Ability to pause and rewind live radio.*
- *Easier Network Management.*

2.2.2 Satellite Broadcasting

[9] Satellite radio is defined by the International Telecommunication Union (ITU)'S ITU Radio Regulations (RR) as a broadcasting-satellite service. The satellite's signals are broadcast nationwide, across a much wider geographical area than terrestrial radio stations. This type of transmission used when high coverage is required.

Advantages of satellite Broadcasting:

- *Transmitted signal is available to a wider coverage without the use of reginal and central stations.*
- *Takes many years to construct a terrestrial network that will achieve comparable coverage.*

Disadvantage of satellite broadcasting

- *Clear line-of-sight condition between the satellite and a radio receiver has to be provided. In most cases, portable and mobile reception in indoor environments is not possible.*

2.2.3 Webcasting

Is a computer mediated broadcasting over the internet service using streaming technology with audio streaming services such a Pandora radio having gained a lot of popularity. Broadcasters have started investing in webcasting this has been due to the improved network speeds, and the increase of individuals accessing the internet. [12] Stated that broadcasting over the network has the potential to disrupt the established industry offering opportunities for online providers to profit, but has potential to diminish revenues for content providers.

Disadvantage of webcasting

- *[10]In some cases, the major drawback of streaming platforms is their insufficient quality, because in order to listen to high-quality audio content one must purchase a premium account.*

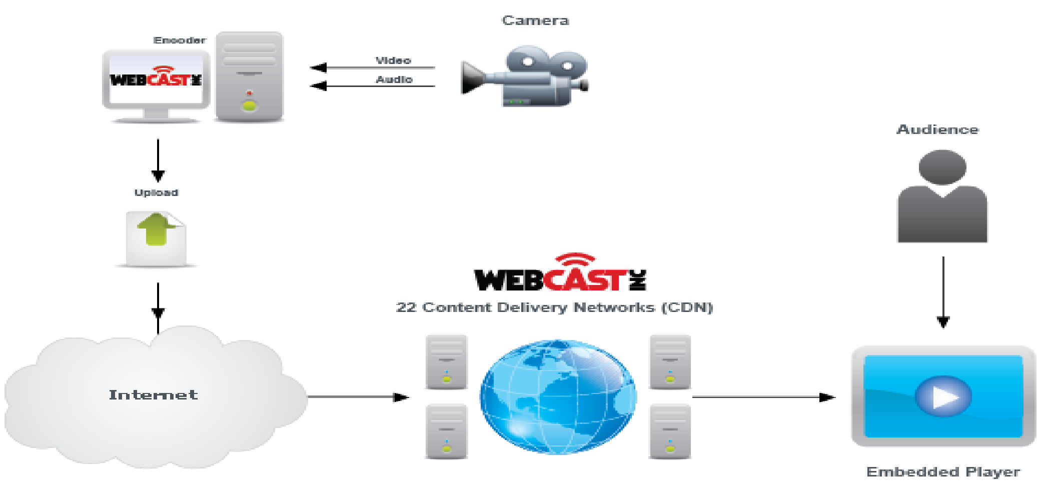


Figure 2.1: Webcasting [10].

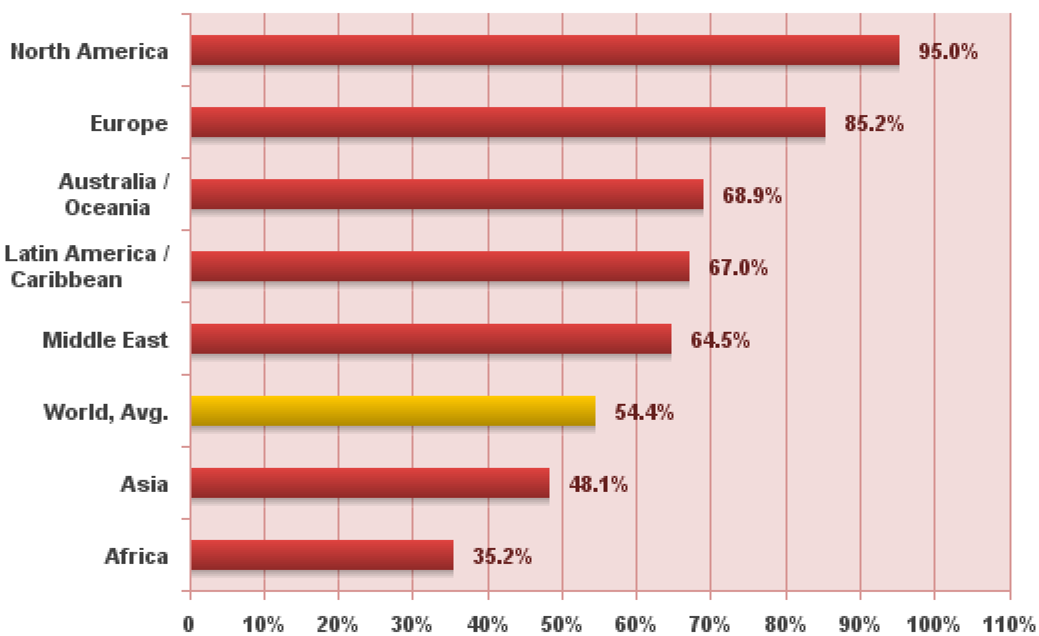
With the increase in internal network bandwidth webcasting has been used by organization such as IBM to create local Radio Stations. Instead of streaming the voice through the internet as shown in Figure 2.1. A local server can be created and content streamed through the organization's internal network also eliminating the CDN. This system can also be used by Universities in Zimbabwe as it is less costly than 2.3.1. And 2.3.2.

2.3 Factors constituting to the rise of webcasting

The rise in webcasting has been largely due to the increase in speed of networks this is largely due to the popularity of the internet, popularity OOT (Over the top) providers, affordability multiplexing on fibre optical communication's, development of packet networks, development of multiprocessing computer chips which have been integrated to a variety of modern devices e.g. laptop's and mobile devices, compression, lower cost of mobile devices.

2.3.1 Popularity of the internet

[11] Defined the Internet as a worldwide interconnection of individual networks operated by government, industry, academia, and private parties. It has eliminated barriers in communication caused by geographical distances, changed the way we do business and helped share knowledge with the internet being largely viewed as the largest library.



Graph 2.1: Internet Penetration Rates December 31 2017 [11].

The internet is always changing with and its revolution in has been due to the innovation in mobile technology with smartphones being largely viewed as a must have in both MEDC (More economically developed countries) and LEDC(Less economically develop countries).The rise in social networks has also contributed to popularity of the internet with this platform's also being used by business's for advertisements.

Table 2.2: Growth Of The Internet [11].

Date	Number of users	% Of World Population
December 1995	16 Million	0.4%
December 2006	1.093 Million	15.7%
December 2011	2.267Million	32.7%
December 2015	3.366Million	46.4%
December 2017	4.157Million	55.4%

2.3.2 Rise of OTT (Over the top Providers)

Over the top providers are companies that provide data or video or audio streaming over the broadband infrastructure for example WhatsApp, Netflix, YouTube, Pandora internet Radio, Spotify etc. [12] say's that this companies eliminate the significant capital investment in building a network, which in turn lowers the barriers to entry into the market. This sites are very popular with majority of internet users using the broadband to primarily access this sites.

For example [12] Netflix does not sell any hardware for playing content. Instead, its Online streaming service is compatible with more than 100 formats on devices such as The Apple iPad and iPod, mobile handheld devices, Roku devices, and game consoles Such as the Microsoft Xbox, Nintendo Wii, and Sony PlayStation. This makes access to your media content than traditional cable provider in which you can only access it on your television.

2.3.3 Fibre Optical Networks

[14] Defines optical Fibre as the medium in which communication signals are transmitted from one location to another in the form of light guided through thin fibers of glass or plastic.French engineer Claude Chappe designed the first optical system in the 1790, this was the optical telegraph. This technology has been developed by engineers and scientist over the past decades with optical being used for a variety of industry such as Military and space, automotive industry and surgery and dentistry etc.

Optical fiber are the backbone of modern networks, this is stated by [12] fiber-optic cabling, multi-core processors, and memory—are the building blocks of modern networks with multiplexing the main reason. Multiplexing allow's multiple light streams to be transported simultaneously. Without multiplexing capacity of networks will be lower than the current capacity.

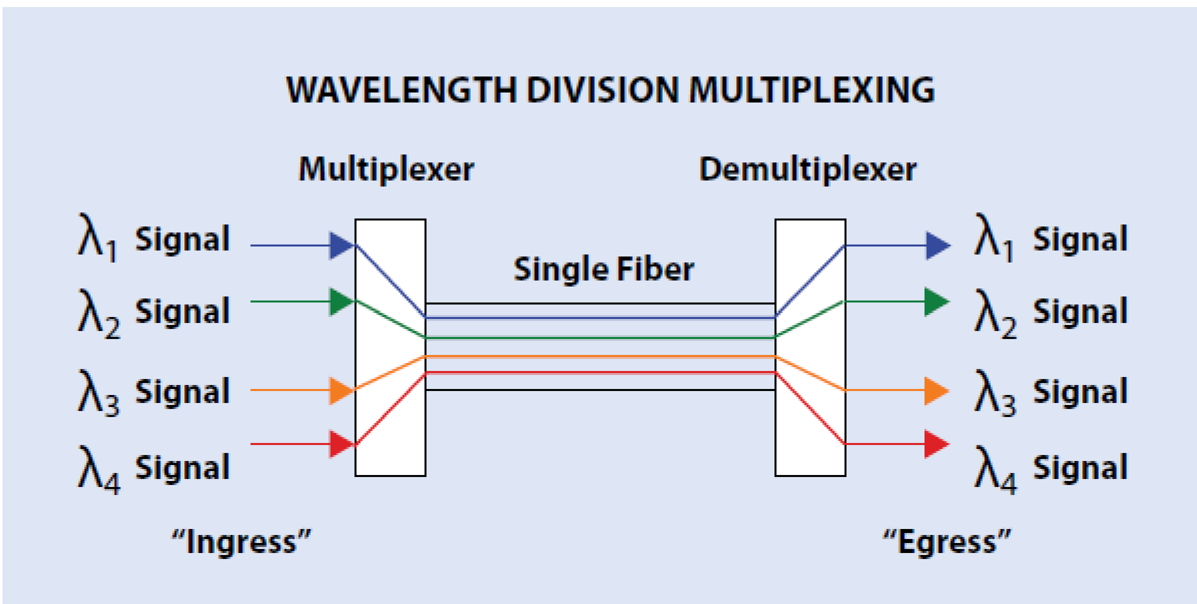
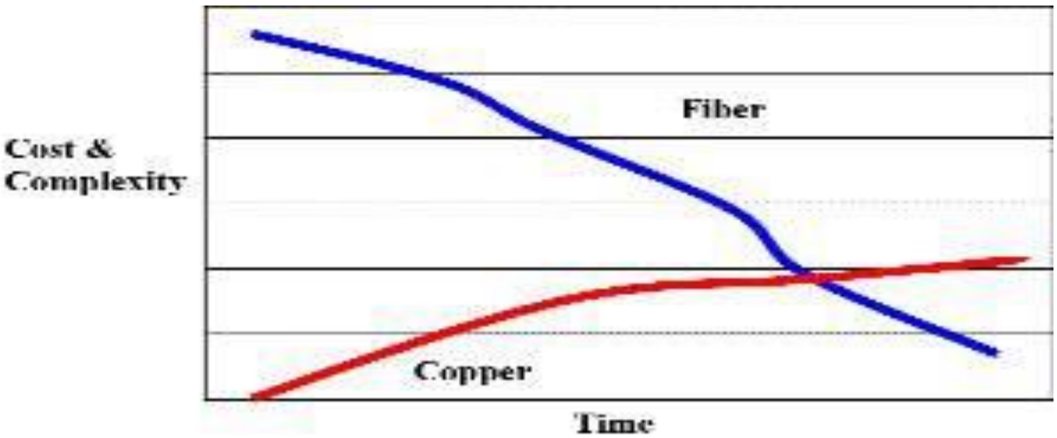


Figure 2.2: Basic WDM Technology Diagram [15].

Over the years the cost of optical fiber has decreased, this has be replayed with optical fibre being the backbone of modern networks with. Today telephone and cable television companies can cost justify installing fiber links to remote sites serving tens to a few hundreds of customers [11].



Graph 2.2: Comparison of Cost and Complexity of Fiber and Copper Over Time [16].

2.3.4 Packet Networks

Majority of data network packets are sent in packets, this is analogues to sending a letter in an envelope. [12] Packet switching was developed by Rand Corporation in 1962 for the United States Air Force and utilized in 1969 in the Advanced Research Projects Agency (ARPANET) of the Department of Defense. ARPANET was the precursor to today's Internet. This networks are more reliable and can handle peak traffic than traditional networks. [12] This was proved in the September 11, 2001 attacks in the United States on the Pentagon in Washington, DC, and the World Trade Center in New York City, the Internet still functioned when many portions of the public voice and cellular networks were either out of service or so overwhelmed with traffic that people could not make calls.

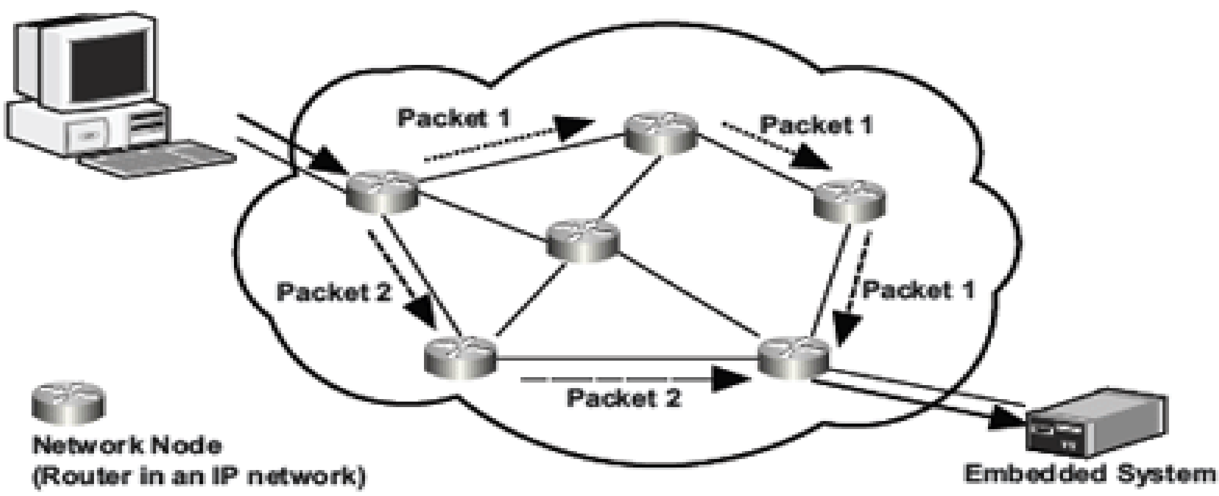


Figure 2.3: Packet Switched Network [17].

If one route on a packet network is unavailable, traffic is rerouted onto other routes.

2.3.5 Multiprocessing Computer Chips

[18] Defines Multiprocessing is the use of two or more central processing units (CPUs) within a single computer system. The term also refers to the ability of a system to support more than one processor or the ability to allocate tasks between them. This chips are being used in various types of network equipment such as mobile devices, Notebooks, Desktop Computers. With the increase in network speed's devices have be able to have high throughput (devices doing more work in less time), high economy of scale, Increased reliability (failure does not halt the system) to handle the new speeds. As network speeds increase Multiprocessing chips have allowed devices to also improve in speed constituting lower latency and higher network throughput.

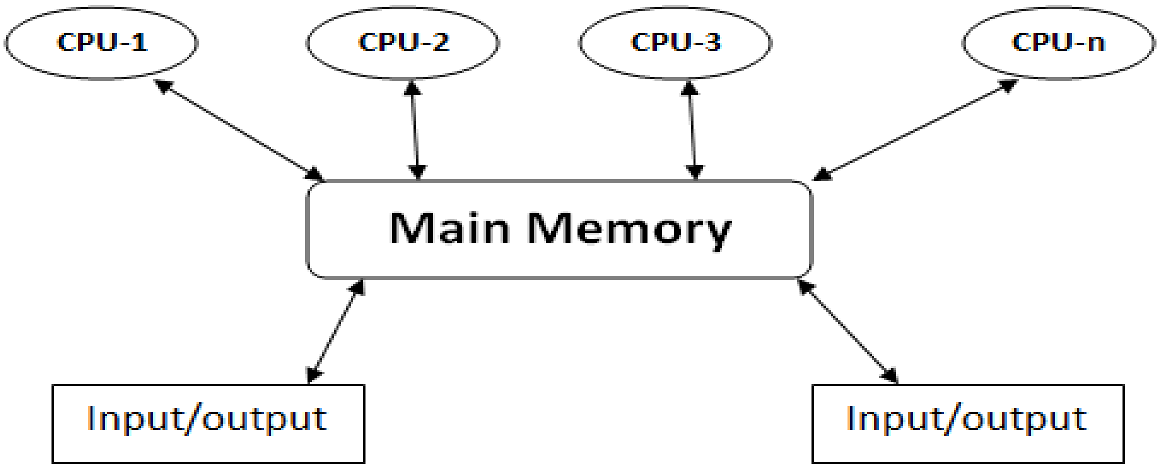


Figure 2.4: Multiprocessing [18]

[19] defined multiprocessing as allowing many operations at the same time with a computer system. For example, we are typing some letter, hearing music, downloading some file from the internet and others.

2.3.6 Compression

Compression is the use of algorithms to decrease the size of voice, data, and video sent over a network. It shrinks the amount audio/video at the transmission end and recreates at it the some quality at the receiving end. Allowing for efficiency of networks without using an enormous amount of bandwidth. It allows a song to be downloaded in seconds rather in minutes. Streaming compression algorithm allows the receiver to have less processing power to receive video, voice, data then the broadcaster.

Speech and television signals are deployed in their original form:

- Analogue: Waves.
- Digital: On and off Switch.

Before transmission on a networks codecs sample speech at different amplitude along the soundwave and converts it to one or zero. Decoder converts the ones and zeros back to analogue sound or video waves. Compression allows codecs ([20] codec stands for Compressor decompressor (or Coder-decoder)) to sample the media to achieve high quality sound. It also allows one to predict the next sound wave to achieve, based on the previous sound to achieve fewer bit rates per second which are transmitted to represent the speech.

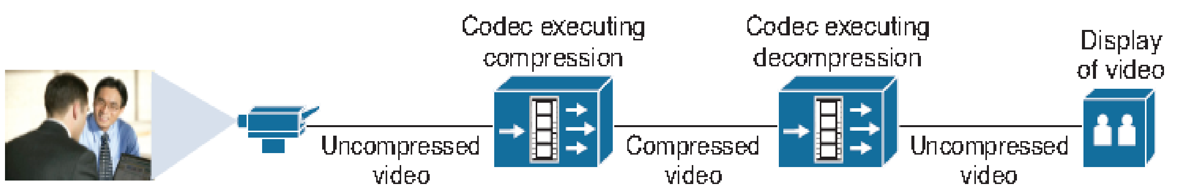


Figure 2.5: Codec Executing Compression [20].

Codec can also referred to video formats.

Table 2.3: Difference Between Compression And Container

Codec(Compression)	Container(File Format)
DV/DVCPRO	MOV
SORENSON SPARK	FLV
On2 VP6-S	MP4V,M4V
H.264	3GP,3G2(mobile)
MP3,AAC(Audio)	MP3,M4A,MP4

Table 2.3 Example: I am creating a FLV file using Sorenson spark codec compressed at 448kbps.

Sample, Sample Format, Sample Rate and Bitrate

- *[21] Defines Sample as a unit of Audio Data.*
- *[21] Defines Sample Format as the number of bits used to describe each sample, e.g. 16bits and 24 bits.*
- *[21]Defines Sample Rate at the number of samples each second e.g. 44,100Hz, 48,000Hz.*
- *[21]Defines as the number of bits per given time second.*
  *Sample Format * Sample Rate=Bit rate*
  *16bits*44100 kHz = 705 kb/s (bitrate for audio CD)*

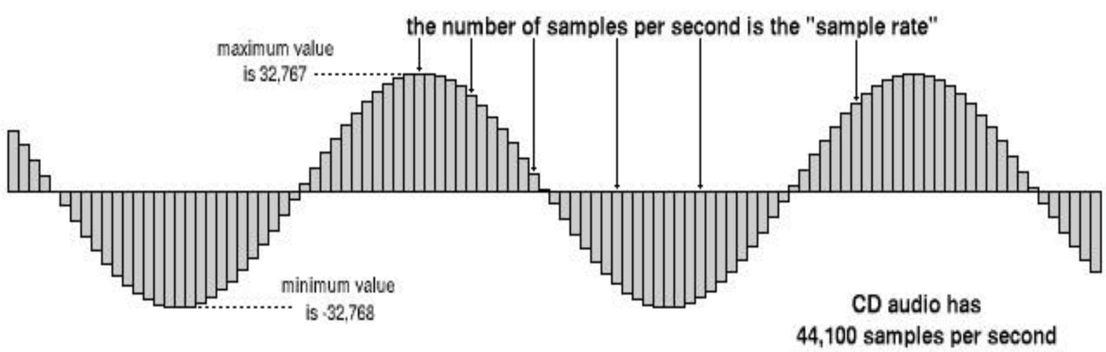


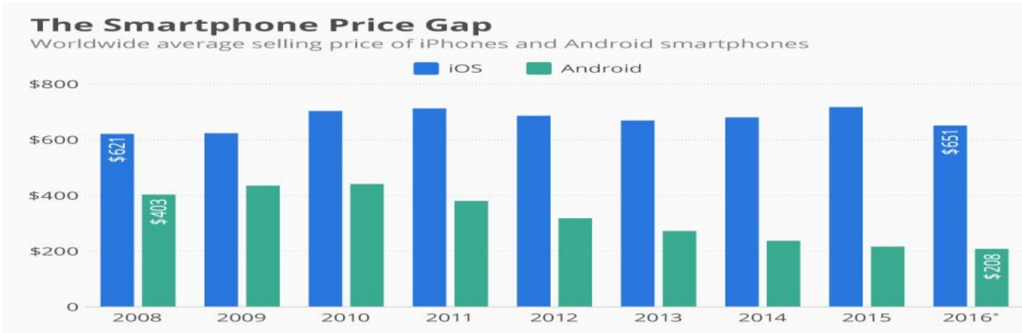
Figure 2.6: Sample Rate [22]

Sample Rate and Sample Format determine sound quality.

[12]Adaptive bitrate software or servers that's dynamically alters the speed of video streams to match consumer and bandwidth. Providing of more consistent Video/Audio Stream.

2.3.7 Lower Cost of Mobile devices

[12] Many experts see the greatest potential for expansion in mobile services in developing countries due to prices of internet-enabled handsets decrease and mobile networks are upgraded.



Graph 2.3: [23] Selling Price Of Smartphones

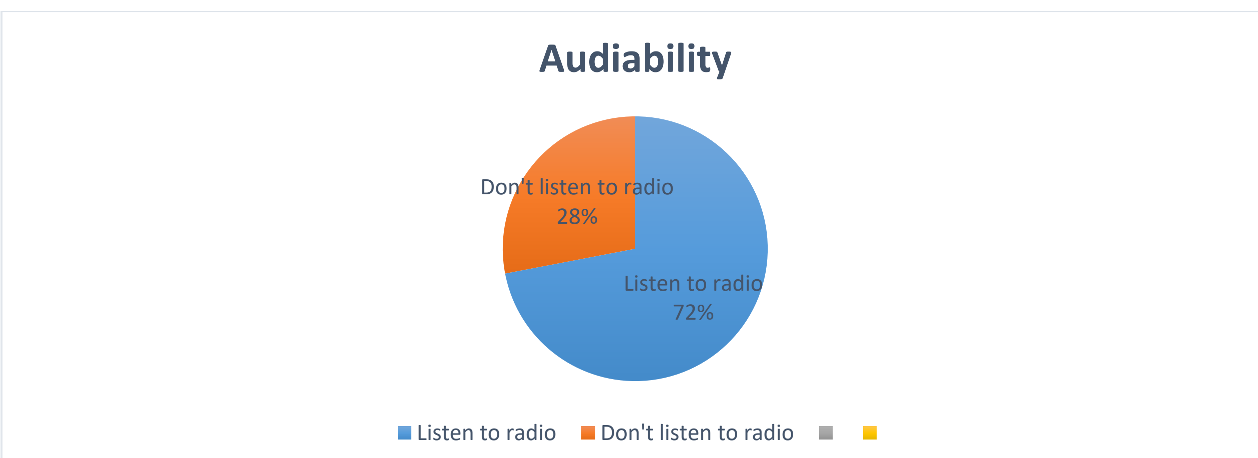
From Figure 2.5 it is noted that the price of android device since 2009 has with iOS devices fluctuating at the same level.

2.4 Case study on user expectation [23]

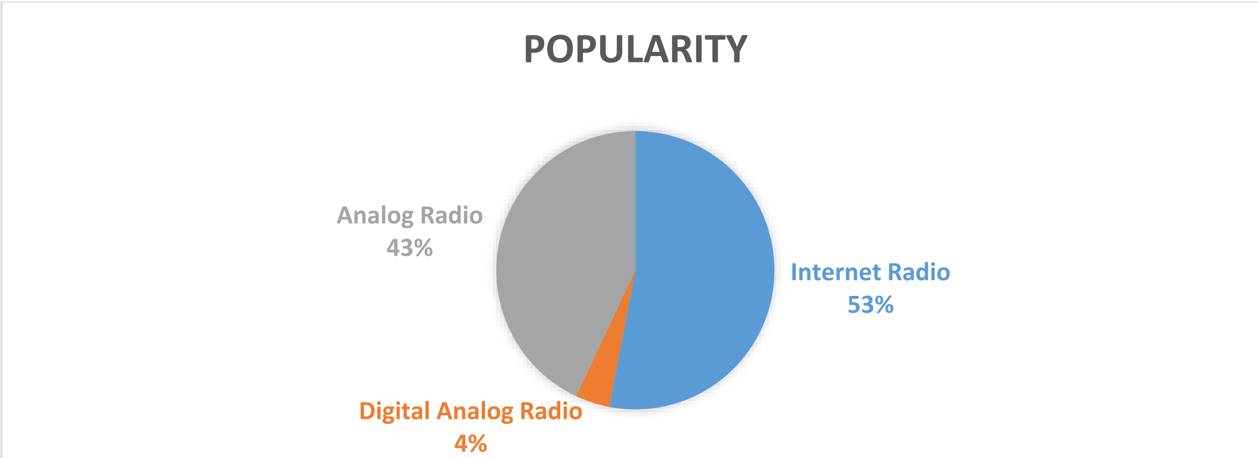
A case study of user expectation for the three types of electronic media including radio, television and internet.

In April 2015 [24] conducted A survey of students representing a group of young people between 18-25 years old. What are their particular needs and expectations related with popular electronic media, especially analog and digital radio transmission?

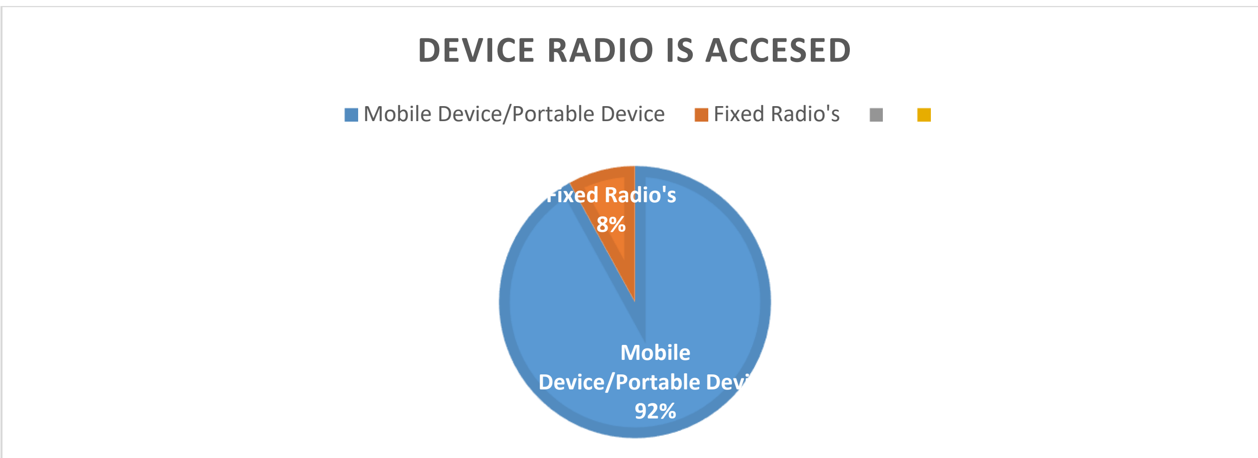
Students filled questionnaire and results of the study are shown below:



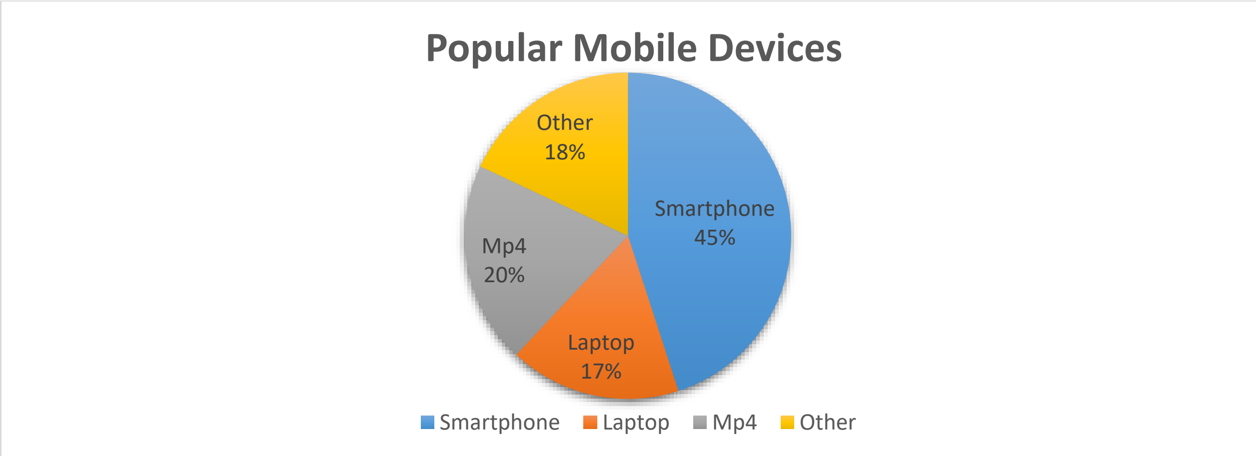
Graph 2.4: Audibility Of People Who Listen To Radio



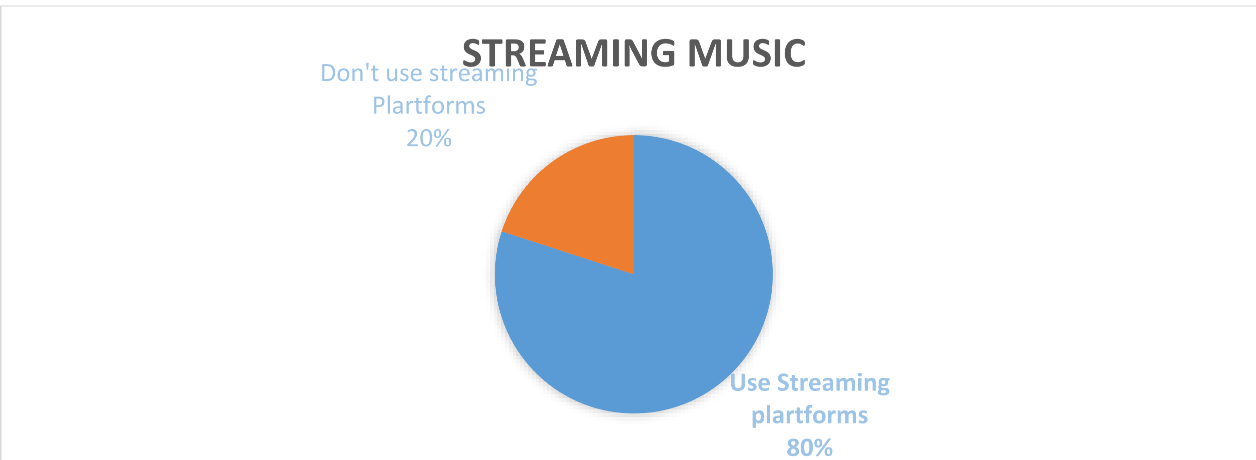
Graph 2.5: Popularity Of Different Types of Radio Broadcasting Techniques



Graph 2.6: How Many People Listen On Mobile Devices



Graph 2.7: Most Popular Mobile Devices



Graph 2.8: Listening To Music On Streaming Platforms

[24] Broadcasters will always aim to use the best possible means to reach the audience in the most effective way. The above results from [24] survey students show that

1. Students prefer listening to internet radio and streaming there content.

2. Students prefer to listen to device on mobile devices with smartphones the most prefers.

3. A large % of students listen to radio.

2.5 Intranet

An intranet is the use of web technology for the sole dedicated use of single site and multisite organizations. Intranets are a way to distribute information, software and other services within an organisation using web like software tools. [12] State that in addition, intranets at global organizations often mimic web functions by providing social networking functions.

2.5.1 Osi Reference Model

Developed in the 1970s by the International Organization for Standardization the Open Systems Interconnection (OSI) architecture, which defines how equipment from Multiple vendors should interoperate. Architecture is the ways that devices in Networks are connected to one another

Table 2.4: OSI Reference Model Layer's

Layer Name /Number	Description	Protocol's
Physical Layer 1	Defines media type.	Wi-Fi
Data Link Layer 2	Addressing information.	IEE802.3, Frame Relay.
Network Layer 3	Routing layer.	Internet Protocol.
Transport Layer	Differentiate Between types of content.	TCP (Transmission control protocol).
Session Layer	Manages the actual dialog of sessions.	H.323.
Presentation Layer	Presentation of information Screen.	Extensible Markup Language (HTML).
Application Layer	Application itself plus specialized Service.	HTTP (Hyper Text Mark Up Language).

2.5.2 Network Protocols

Physical layer /Data link layer protocols

- **IEE802.11 Wi-Fi (Wireless fidelity)** – Wi-Fi is a wireless technology used for LAN (Local Area Network). Wi-Fi can use the 2.4 gigahertz ultra-high frequency and 5.8 gigahertz super high frequency radio band.[24] defines IEEE 802.11 is a set of media access control (MAC) and physical layer (PHY) specifications for implementing wireless local area network(WLAN) computer communication in the 900 MHz and 2.4, 3.6, 5, and 60 GHz frequency bands.

Table 2.5: IEEE 802.11 Standards

Protocol	Release Date	Operational Frequency	Data Rate (Max)	Range (Indoor)	Range (Outdoor)
Legacy	1997	2.4-2.5 GHz	1 Mb/s	?	?
802.11a	1999	5.15-5.35/5.47-5.725/5.725-5.875 GHz	25 Mb/s	25 meters	75 meters
802.11b	1999	2.4-2.5 GHz	5.5 Mb/s	35 meters	100 meters
802.11g	2003	2.4-2.5 GHz	25 Mb/s	25 meters	75 meters
802.11n	2007	2.4 GHz or 5 GHz bands	200 Mb/s	50 meters	126 meters

- **Ethernet Physical Layer-**[25] defines the Ethernet physical layer is the physical layer functionality of the Ethernet family of computer network standards. The physical layer defines the electrical or optical properties of the physical connection between a device and the network or between network devices.

Speed	Common Name	Informal Name	Formal IEEE Name	Cable and Max. Length
10 Mbps	Ethernet	10BASE-T	802.3	Copper, 100 m
100 Mbps	Fast Ethernet	100BASE-T	802.3u	Copper, 100 m
1000 Mbps	Gig Ethernet	1000BASE-LX	802.3z	Fiber, 5000 m
1000 Mbps	Gig Ethernet	1000BASE-T	802.3ab	Copper, 100 m
10 Gbps	10 Gig Ethernet	10GBASE-T	802.3an	Copper, 100 m

Figure 2.7: Ethernet Standards [27]

Network layer

- **(IPv4)Internet protocol version 4-** [27] defines Internet Protocol version 4 (IPv4) as the fourth version of the Internet Protocol (IP). It is one of the core protocols of standards-based internetworking methods in the Internet, and was the first version deployed for production in the ARPANET in 1983. It has 32 bit addresses and reserves special address blocks for private networks. [27] Written in the dot-decimal notation, which consists of four octets of the address expressed individually in decimal numbers and separated by periods e.g. 127.0.0.1.

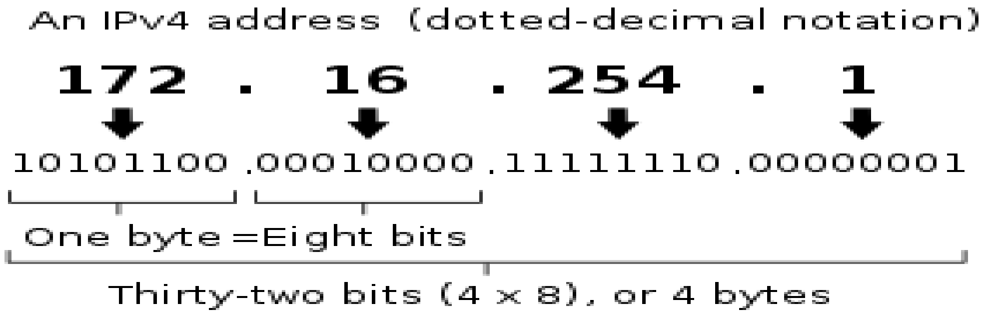


Figure 2.8: IPv4 Address[27]

Transport layer

- **TCP(Transmission Control protocol)** –Main protocols on the internet suite.TCP provides reliable, ordered, and error-checked delivery of a stream of octets (bytes) between applications running on hosts communicating via an IP network [28]. Used by Internet applications e.g. World Wide Web, email, remote administration, and transfer rely on TCP.
- **UDP(User Datagram protocol)** – Core protocols on the internet suite.[29] With UDP, computer applications can send messages, in this case referred to as datagrams, to other hosts on an Internet Protocol (IP) network. Prior communications are not required in order to set up communication channels or data paths.

Session layer

Table 2.6: Difference Between TCP And UDP

TCP	UDP
Reliable	Unreliable
Connection-Oriented	Connectionless
Segment Retransmission	No retransmission

Flow control Through windowing	Windowing
Segment sequencing	No sequencing
Acknowledgment Segments	No Acknowledgement

- **RTP Control Protocol-** [30] The RTP Control Protocol (RTCP) is a sister protocol of the Real-time Transport Protocol (RTP). Its basic functionality and packet structure is defined in RFC 3550. RTCP provides out-of-band statistics and control information for an RTP session. It partners with RTP in the delivery and packaging of multimedia data, but does not transport any media data itself.
- **SSH (Secure Shell) –** [31] Secure Shell (SSH) is a cryptographic network protocol for operating network services securely over an unsecured network.[1] The best known example application is for remote login to computer systems by users.

Presentation Layer

- **XML (Extensible Markup Language) -** XML was designed to store and transport data, designed to be both human- and machine-readable. Used for distributing data over the internet.

Application layer

- **RTP (Real Time Transport Protocol) -** [32] The Real-time Transport Protocol (RTP) is a network protocol for delivering audio and video over IP networks. RTP is used extensively in communication and entertainment systems that involve streaming media, such as telephony, teleconference applications including WebRTC, television services and web-based push-to-talk features.
- **Real Time Streaming Protocol (RTSP**) – [33] is a network control protocol designed for use in entertainment and communications systems to control streaming media servers. The protocol is used for establishing and controlling media sessions between end points. Clients of media servers issue VCR-style commands, such as play, record and pause, to facilitate real-time control of the media streaming from the server to a client (Video on Demand) or from a client to the server (Voice Recording). Uses the Real-time Transport Protocol in conjunction with Real-time Control Protocol for media stream delivery
- **Hypertext Transfer Protocol (HTTP)** – [34] is an application protocol for distributed, collaborative, and hypermedia information systems HTTP is the foundation of data communication for the World Wide Web.

Table 2.7: Comparing RSTP Streaming And HTTP Steaming

RTSP	*HTTP*
Longer Movie data rates than connection.	*Higher Data Rates, better quality.*
No File Stored, better for long movies.	*File Download replay is easy.*
Can’t stream all data types.	*Can stream At QT data types*
Can do live.	*Can’t do live.*

From the comparison above. With the radio having to go live RSTP in Conjunction with RTP and RTCP is the best protocol to use when designing the intranet Radio.

REFERENCE LIST

[1] W.DeJong. (2002.). “The role of mass media campaigns in reducing high-risk drinking among college students. Journal of Studies on Alcohol. [On-line]. 14(1), pp. 182-192. Available: http://dx.doi.org/10.15288/jsas.2002.s14.182 /, 2002 [April, 26 2018].

[2] Bahaeldin Ibrahim and Naveen Mishra. “College Radio as a Mechanism for Participatory Learning: Exploring the Scope for Online Radio Based Learning Among Undergraduates.” Higher Learning Research Communications, vol.6, pp. 22, March. 2016.

[3] N.A. “What languages are spoken in Zimbabwe: https://www.discoverafrica.com/faq/what-languages-are-spoken-in-zimbabwe/ [April, 25 2018].

[4] Mehmood & Hameed, Y. M. Y. (2014.). “Role of radio in imparting education at Allama Iqbal Open University Pakistan and Al-Qudus Open University Jordan. Asian Journal of Social Sciences & Humanities. [On-line]. 3(1), pp. 261-366. Available: Retrieved from http://www.ajssh.leena-luna.co.jp/2002 [April, 26 2018].

[5] Volda Elliott *et al*. “The effectiveness of Interactive Radio Instruction (IRI) within selected Primary Schools in Region Number Four (4).” Social Science Learning Education Journal, vol.2, pp. 22, August. 2017.

[6] Ho, J., & Thukral. “Tuned in to student success: Assessing the impact of interactive radio instruction for the hardest-to-reach.” Journal of Education for International Development, vol.4, pp. 34-51.2009.

[7] Aqrabawi, T., Zaidah, S., & Kuttab, D. “Community radio for development in Jordan with specific reference to AmmanNet Radio. World Association of Community Radio broadcasters (AMARC). Available: http://www.amarc.org [2006].

[8] Mefalopulos, P. “D Theory and practice of participatory communication: The case of the FAO project "Communication for Development in Southern Africa".” Doctoral dissertation, University of Texas, Austin, TX, 2003

[9] N.A. “Satellite radio: https://en.wikipedia.org/wiki/Satellite_radio/ [April, 26 2018].

[10] N.A.“http://webcastinc.com/what-is-webcasting/live-webcast-how-to-stream./ [April, 25 2018].

[11] N.A. “https://www.internetworldstats.com/emarketing.htm / [April, 26 2018].

[12]A.Z. DODD, The Essential Guide to Telecommunications Fifth Edition, 5th ed. Harlow: Prentice Hall, 1998.

[13] Jürgen Meffert and Niko Mohr “Overwhelming OTT: Telcos’ growth strategy in a digital world: https://www.mckinsey.com/industries/telecommunications/our-insights/overwhelming-ott-telcos-growth-strategy-in-a-digital-world/ [April, 26 2018].

[14] Jim Hayes. Fibre Technician Manual. N/A, Delmar Publishers, 1996, pp. 1-12.

[15] N.A. “http://www.fiber-optic-transceiver-module.com/wavelength-division-multiplexing-wdm-technology.html [April, 26 2018].

[16] N.A. “http://www.networkmagazineindia.com/200205/krone2.shtml/ [April, 26 2018].

[17] Christian Legare, Micrium “https://www.embedded.com/print/4199383 [April, 26 2018].

[18] N.A. “https://en.wikipedia.org/wiki/Multiprocessing [April, 26 2018].

[19] N.A. “https://rsktutors.blogspot.com/2016/10/multiprocessing.html [April, 26 2018].

[20] N.A, Cisco Video and Telepresence Architecture Design Guide: Cisco, N.D.

[21] N.A. “https://docs-old.fedoraproject.org/en-US/Fedora/15/html/Musicians_Guide/sect-Musicians_Guide-Sample_Rate_and_Sample_Format.html [April, 26 2018].

[22] N.A. “http://www.cras.edu/bit-depth-sample-rate-affect-loudness-measurements/ [April, 26 2018].

[23] N.A. “https://www.statista.com/chart/4954/smartphone-average-selling-prices/ [April, 26 2018].

[24] Przemysław Gilski and Jacek Stefański. “Can the Digital Surpass the Analog: DAB+ Possibilities, Limitations and User Expectations?.” INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, vol.62, pp. 353-361.Nov, 2016.

[25] N.A. “https://en.wikipedia.org/wiki/802.11 [April, 26 2018].

[26] N.A. “http://www.learncertification.com/study-material/identify-the-appropriate-media-cables-ports-and-connectors-to-connect-cisco-network-devices-to-other-network-devices-and-hosts-in-a-lan html [April, 26 2018].

[27] N.A. “https://en.wikipedia.org/wiki/IPv4 [April, 26 2018].

[28] N.A. “https://en.wikipedia.org/wiki/Transmission_Control_Protocol [April, 26 2018].

[29] N.A. “https://en.wikipedia.org/wiki/User_Datagram_Protocol [April, 26 2018].

[30] N.A. “https://en.wikipedia.org/wiki/RTP_Control_Protocol [April, 26 2018Ch].

[31] N.A. “https://en.wikipedia.org/wiki/Secure_Shell [April, 26 2018].

[32] N.A. “https://en.wikipedia.org/wiki/Real-time_Transport_Protocol [April, 26 2018].

[33] N.A. “https://en.wikipedia.org/wiki/Real_Time_Streaming_Protocol html [April, 26 2018].

[34] N.A. “https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol [April, 26 2018].

CHAPTER 3

3 REARSERCH METHODOLOGY & TECHNIQUES

3.1 Introduction

This section provides the relationship and interconnection between the hardware components that are listed in Section 3.4 to produce the design in Fig 3.1 to meet the aim of the dissertation.

3.2 Assumptions

The student designed a prototype of an intranet radio, Assumption's made in the designing of the prototype where as follows:

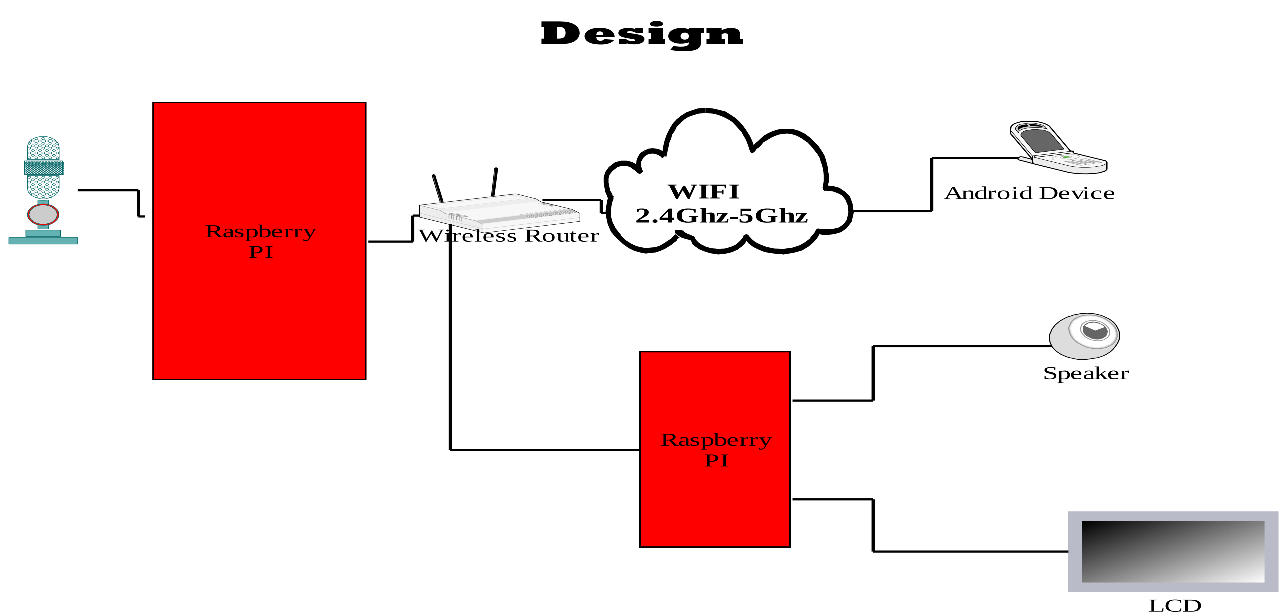


Figure 3.1: Design Of Prototype

Assumptions:

1. *Interfacing the components in Figure 3.1 demonstrates an intranet.*
2. *Raspberry Pi connected to the Mic is Streaming server with encoder.*
3. *Raspberry Pi interfaced with speakers and Lcd demonstrates a Radio.*

3.3 System Overview

The prototype Consists of Three Main Building blocks.

- A streaming Server.
- A network Router.
- An internet radio and android device for reception.

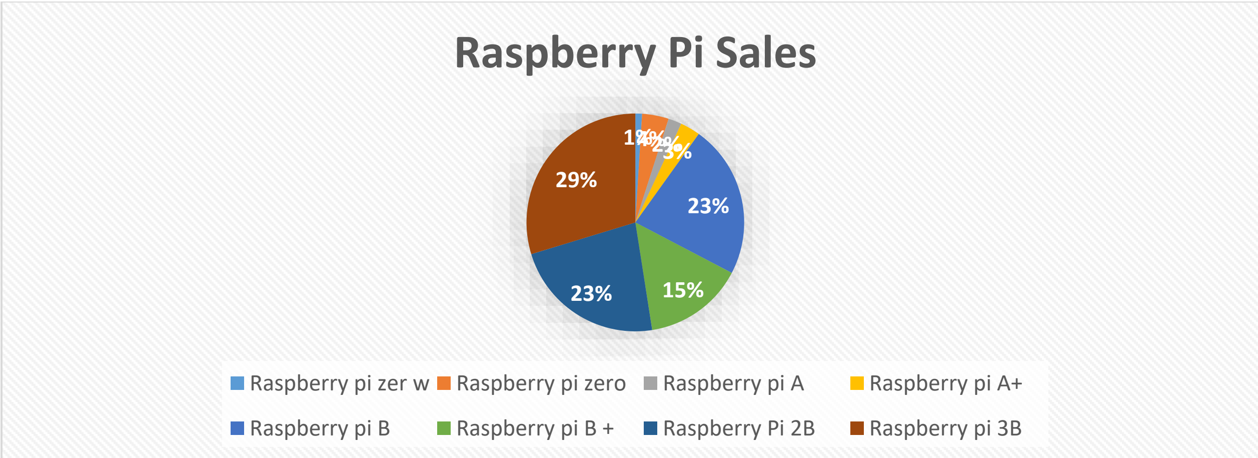
A raspberry pi which will receive voice signal from a microphone, it will then convert the analog signal to digital format. A Software called ffmpeg will encode the voice which is now in digital format and send it to ffserver. Ffserver will stream the voice to the wireless router. The router should then route the stream to the Internet Radio and the android device. The internet radio and android application will then translate the stream from the server into the voice to be heard by the listener.

3.4 Specifications of components used

3.4.1 Raspberry Pi

Raspberry Pi Foundation is a charity foundation in the United Kingdom to empower people all over the world in order to understand and shape the digital world. This Objective is being achieved by the production of low cost computers.

The first Raspberry Pi single board computer was developed in 2011 and sold in 2012 with 6 models being produced since the raspberry pi zero w.



Graph 3.1: Breakdown of Raspberry Pi Sales [1]

Raspberry Pi 3B replaced the raspberry pi 2B in 2006. It is the earliest version of the 3rd generation of the raspberry pi. It is 10* faster than the first generation of raspberry pi. It also has wireless LAN and Bluetooth connectivity.

Figure 3.2: Raspberry Pi 3B

Specifications

Table 3.1: Raspberry Pi Specifications

Specifications	
Processor	Broadcom BCM2387 chipset. 1.2GHz Quad-Core ARM Cortex-A53 802.11 b/g/n Wireless LAN and Bluetooth 4.1 (Bluetooth Classic and LE)
GPU	Dual Core Video Core IV® Multimedia Co-Processor. Provides Open GL ES 2.0, hardware-accelerated OpenVG, and 1080p30 H.264 high-profile Decode.
Memory Capacity	1GB DDR2
Operating System	Linux operating system or Windows 10 IoT
Physical Dimensions	85 x 56 x 17mm
Power	Micro USB 5V, 2.5A

Table 3.2: Raspberry Pi Connectors

Connectors	
Ethernet	10/100 BaseT Ethernet socket
Video Output	HDMI (rev 1.3 & 1.4 Composite RCA (PAL and NTSC)
Audio Output	Audio Output 3.5mm jack, HDMI USB 4 x USB 2.0 Connector
GPIO Connector	40-pin 2.54 mm (100 mil) expansion header: 2x20 strip Providing 27 GPIO pins as well as +3.3 V, +5 V and GND supply lines
Camera Connector	15-pin MIPI Camera Serial Interface (CSI-2)
Display Connector	Display Serial Interface (DSI) 15 way flat flex cable connector with two data lanes and a clock lane
Memory Card Slot	Push/pull Micro SDIO

3.4.2 Cat 5e Universal twisted Pair

[3] Describes twisted pair cables cabling is a type of wiring in which two conductors of a single circuit are twisted together for the purposes of improving electromagnetic compatibility. Reduces electromagnetic radiation, cross talk between neighboring pairs and improves rejection of electromagnetic interference.

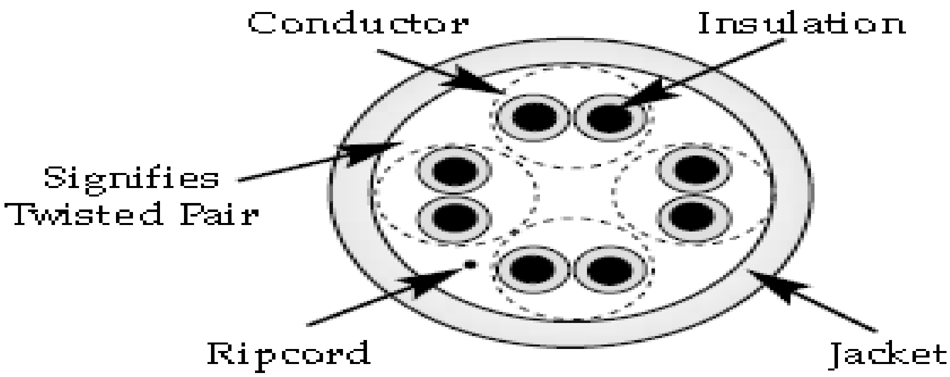


Figure 3.3: UTP [4]

Cat 5e UTP has a bandwidth of 100 MHz being applied to 100Base-TX/1000 Base-T Ethernet. It has the some construction as a Cat5 with better testing standards. It is common for local Area networks

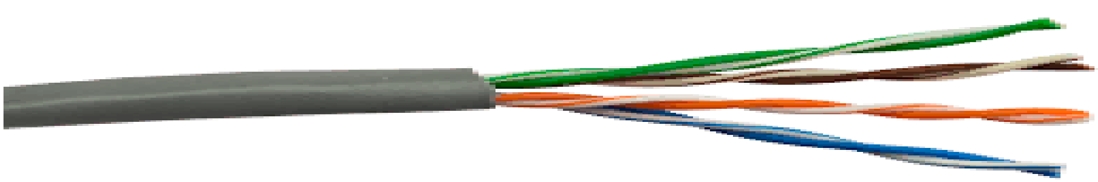


Figure 3.4: CAT 5E UTP [5]

3.4.3 Huawei: Home Gateway Model: HG630 V2

[6] Describes the HG630 is a new generation wireless gateway that supports very-high-data-rate digital subscriber line 2 (VDSL2). The HG630 is designed for mid-range and high-end Users using high bandwidth services, such as HD video. It comes with a vectoring function that keeps the transmission rate at 100mb/s within the range of 300m away, without this function the rate decreases to 70mb/s when a user is 300m away.

Can provide wireless transmission rates of up to 300mb/s making it ideal for HD video streaming. This is due to the (MIMO) Technologies using the 802.1n.

Table 3.3:Functional Feature's [6]

Functional Feature's	Description
High-bandwidth VDSL2 Upstream Link	Comes with a high performance VDSL2 network processor.
WLAN Function	Compliant with 802.11b, 802.11g and 802.11n (2.4GHz-2.4835GHz).
WPS Function	Provide secure wireless function between computer and router by pressing the wps button.
Routing Function	Compliant with PPP dialer and a built in DHCP allowing it access to multiple users.
IPv6 Function	Supports IPv4and IPv6
Flexible QoS Policies	Multiple levels Of traffic classification insuring quality video and audio.
Standardized TR-069 Management	Low operation and maintaince cost.
Convenient and Secure Management and Maintenance	Allows remote management on a Web based configuration utility.

Table 3.4: Technical Specification's [6]

Technical Specifications	Description
DSL Interface	➢ Multiple DSL Standards
WLAN Interface	➢ Compliant with 802.11b, 802.11g and 802.11n (2.4 GHz ~ 2.4835GHz) ➢ WPS 2.0 (PBC mode and PIN mode) ➢ Supports DQPSK, DBPSK, CCK, OFDM, BPSK, QPSK, 16-QAM and 64-QAM wireless modulation method. ➢ Multiple SSIDs and SSID hiding ➢ WPA1.0 and WPA2.0 security ➢ 64/128 digits WEP encryption ➢ TKIP and AES encryption

	➢ WMM ➢ Supports enable or disable the WLAN function by press WLAN button or config the web-based utility ➢ WLAN Rates: • 802.11b: Up to 11 Mbit/s • 802.11g: Up to 54 Mbit/s • 802.11n(with a 2×2 antenna used):Up to 300.0 Mbit/s
USB Interface	➢ Functions as a USB Host 2.0 interface for USB storage devices or printers. ➢ Accessing a portable storage device through FTP server
Security Features	➢ Powerful wireless network security ➢ IP/MAC filtering ➢ URL filtering ➢ ACL access control ➢ Prevents DoS attacks such as LAND, SYN flooding, ICMP Smurf, Ping of Death, Ping Sweep, Teardrop, Unreachable, TCP/UDP PortScan and ICMP Redirection.
Routing & Bridged Features	➢ Supports IPv6 • IPv4 and IPv6 dual-stack • DS-Lite Tunnel • SLAAC ➢ NAT and ALG expansion ➢ DHCP Server/Client ➢ DNS Relay/Client ➢ IGMP proxy and IGMP snooping ➢ DMZ ➢ UpnP ➢ SNTP ➢ Port mapping ➢ RIP V1&V2 ➢ Bridging between the WAN port and the LAN port
QoS Features	➢ Supports 802.1p and 802.1q ➢ Agile QoS Strategy ➢ Rich of stream classification strategy
Network Management	➢ Supports TR-069 ➢ Views system logs ➢ Two levels of web access control ➢ Prevents improper upgrades ➢ Upgrade through TR-069 ➢ Remote and local web configuration and management

	➢ Backing up and restoring the configuration
Power Supply Specifications	➢ Entire-device power supply: 12 V DC, 1 A ➢ Entire-device power consumption: < 12 W
Environmental Specifications	➢ Ambient temperature for operation: 0°C to 40°C (32°F to 104°F) ➢ Relative humidity for operation: 5% to 95%, non-condensing

3.4.4 16 X 2 Module Hd44780 Display Green

This is a 16*2 LCD display module is a high quality display with a yellow backlight, wide viewing angle and high contrast. This are typically used for copiers, fax machines and lasers.

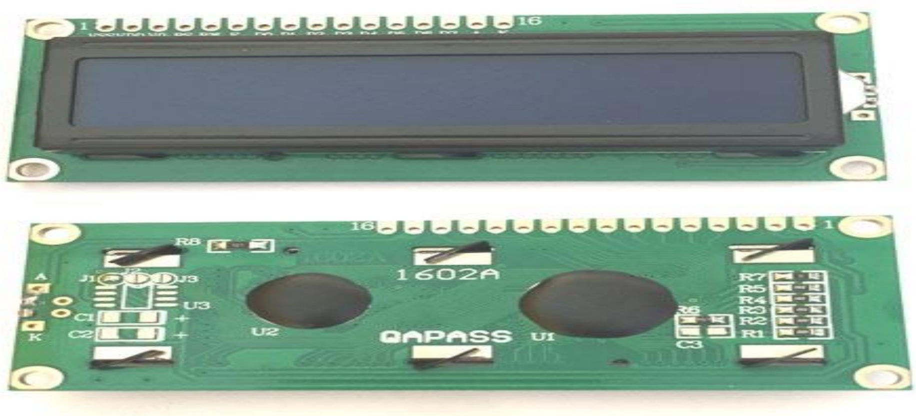


Figure 3.5: Hd44780 LCD Display [7]

Table 3.5: Terminal Functions [7]

Pin	Name	Descriptions
1	VSS	0V Power Supply
2	VDD	Positive Power Supply
3	V0	LCD Contrast
4	RS	Register Select
5	R/W	Read or Write
6	E	Enable
7-14	DB0-DB7	Data Bus (0-7)
15	BLK	LCD Backlight(Ground)

16	BLA	LED Backlight (5v)

Table 3.6: LCD Characteristics [8]

Item	Symbol	Min	Type	Max
Supply Voltage	VDD-VO	-	3.0	-
Input Voltage(V)	VDD	3.1	3.3	3.5
Supply Current	IDD	-	1.5	2.5
Operating Temperature	VOP	0	-	50

3.4.5 Model P160 16mm Rotary Potentiometer

[9] Describes the 10K potentiometer as a two-in-one, well in a breadboard or with a panel. It's a fairly standard linear taper 10K ohm potentiometer, with a grippy shaft.

Figure 3.6: 10K Potentiometer [9]

Table 3.7: Electrical Specification's [10]

Specification	Range
Resistance Range,	Ohms 500-1M
Standard Resistance	Tolerance ± 20%
Residual Resistance	20 ohms max.
Power rating Input Voltage,	Maximum 200 Vac max.
Power Rated,	Watts 0.2W- B taper, 0.1W-others
Dielectric Strength	500Vac, I minute

Insulation Resistance	Minimum 100M ohms at 250Vdc
Sliding Noise	100mV max.
Actual Electrical Travel	260

Table 3.8: Mechanical And Environmental Specification's [10]

Specification	Range
Total Mechanical Travel	300°± 10°
Static Stop Strength	90 oz-in
Rotational Torque	0.5 to 1.25 oz-in
Operating Temperature Range	-20°C to +70°C
Rotational Life	100,000 cycles

3.4.6 USB 2.0 Virtual 7.1 Channel Sound Card Adapter

[11] Adds external microphone, speaker and/or headphone ports, as well as volume and mute controls, to your computer or laptop via USB port. The Sound Card plugs into your computer or laptop's USB port to provide easy-to-access audio-in and audio-out ports.

Figure 3.7: 7.1 Channel Sound Card Adapter [11]

Table 3.9: Sound Card Specifications [12]

Specification	Value
LED Indicators	Red (Speaker/Mic Mute), Green (Active)
Operating Temperature Range	32 to 104 F (0 to 40 C)
Storage Temperature Range	14 to 131 F (-10 to 55 C)
Relative Humidity	5% to 90% RH, Non-Condensing
BTUs	1.7 BTU/Hr
Power Consumption (Watts)	0.5
Ports	2
USB Specification	USB 2.0 (up to 480 Mbps)

Audio Specification	7.1 Surround Sound
Driver Required	No

3.4.7 INTEX IT-320 Computer Multimedia Speaker

Silver black plug and play multimedia speakers with an attractive look and superior performance. This speakers have a low distortion rate, high power output, and an excellent frequency response.

Table 3.10: Speaker Specification's [13]

Specification	Value
Power Supply	AC 220-240,50/60Hz
Speaker Size	5.72cm(2.25)
Power Output	2 Watt RMS per Channel
Impedance	8 Ohms

3.4.8 Aneex Auricular JY-HT008 Microphone with noise reduction

This is a high quality headphone with a built in microphone, extra base sound and a noise reduction function.

Table 3.11: Microphone Specification's [14]

Specification	Value
Impedance	32Ohms
Sensitivity	112db
Frequency Range	20-20.00-0Hz
Output Power	40mW
Cable Length	1.6M

3.4.9 Momentary Contact Push button

Push button is a small switch that completes an electric circuit when used. Made of non-conducting plastic. When it's on, a small metal spring inside makes contact with two wires which allows the circuit to be completed. When it's off, the spring retracts, contact is interrupted and the circuit is not completed. Momentary switches work's as long as one press on them e.g. buttons on a phone or calculator.

3.4.10Android Device

Android is an open source operating system based on Linux Kernel[15]. Android OS shares more than half the global market this is largely due to the free patent and the affordability of the device using this operating system.

Advantages of Android:

- High opening.
- Off operator restraint.
- Self-defined application.
- Easily use of the Google application.
- Low price.

3.5Software Packages Used

3.5.1 FFserver & Ffmpeg

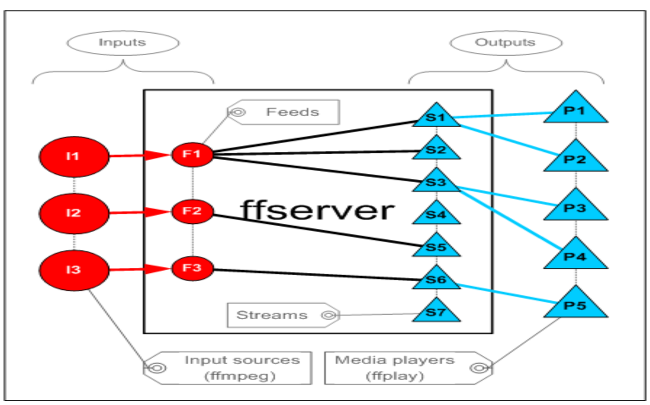


Figure 3.8: Software Configuration [16]

FFserver is a streaming server for both audio and video. It supports several live feeds, streaming from files and time shifting on live feeds streaming them over RTP/RTSP/HTTP.

Ffmpeg is a platform that allows recording, streaming, converting and files, audio and video.

3.5.1 Raspabian Stretch

Raspabian is the Raspberry Pi Foundation's official supported operating system. Raspabian stretch is the latest version of the operating system.

3.5.2 Vlc

Vlc is a free open source cross multimedia player that plays most multimedia files well as DVDs, Audio's and various streaming protocols. Video Lan is a project made by a non-profit organization.

3.5.3 Putty

Putty is Ssh and telnet client. It is an open sourece software.

3.5.4 Vnc Viewer

Vnc Viewer is a screen sharing software that allows you to solve problems and work remotely.

3.5.5 SD card formatter

SD card formatter formats SD memory card, SDHC memory card and SDXC memory card.

3.5.6 Win32disk

Win32disk is designed to write a raw disk image to a removable device or backup a removable device to a raw image file.

3.5.8 Android studio

Android studio Provides tools for building apps on every type of Android device.

3.5.9 Wireshack

Wireshark is a network protocol analyzer. That allows one to monitor a wide range of protocols e.g http, rstp, rtp, udp e.t.c

3.6 Streaming Server

Components used to build the streaming server, are

1. Raspberry pi 3B and peripherals: Which was acting is acting as the dedicated server and encoder. Peripherals include a 5V, 2.1Amp Samsung charger to power the raspberry pi, and a 16GB memory to provide memory to the pi.

2. USB 2.0 Virtual 7.1 Channel Sound Card Adapter: Due to the Raspberry pi not having an audio input jack a usb sound card adapter was used to provide the raspberry pi with an audio input jack.

3. **Annex Auricular JHT008 with sound card:** A Pair of inexpensive headphones was used to provide voice input into the raspberry pi.

4. **Huawei: Home Gateway Model: HG630 V2:** Used to provide network between pi and students PC.

5**. 2*Cat 5e Universal twisted Pair:** Used to connect router to students PC and router to raspberry pi.

6. **HP Pavilion**: Used to provide peripherals to raspberry pi.

Step 1: Installing Raspabian, Connecting device to laptop and connecting pi to internet.

Installation of Raspabian stretch (Open source Raspberry pi Operating system):

1. Connect the 16GB memory card to a SD card adapter and plug into the laptop.
2. A software called SD Card formatter was downloaded and used to format the memory card.
3. The Raspabian stretch disk image and Win32DiskImager was also downloaded.
4. Raspabian stretch was downloaded as a rar file, this had to be extracted by WinZip to obtain the image.
5. Win32DiskImager was used to install raspabian stretch in to the SD Card.

Accessing Raspberry pi GUI (Graphical user interface):

1. Insert the SD card into the SD card slot of the Raspberry Pi.

2. Connect the Samsung charger to the pi and allow the raspberry pi to boot the freshly installed SD card with operating system.

3. After boot, power of pi and remove the memory card and use the SD card reader to connect to pc.

4. Command *echo>H/ssh* was used to allow a putty to allow an ssh connection with raspberry pi.

5. Card should be safely removed from Pc and inserted into the raspberry pi and the raspberry pi booted this time connecting to the wireless router. The wireless router was also connected to the pc using the utp cables.

6. Putty was run and an ssh connection was established with the raspberry pi.

7. Username name for the pi: *pi* and password: *raspberry*.

8. *Sudo raspi-config* was typed to enter the raspberry pi settings.

9. The interface setting's where changed allowing SSH and VNC.

10 The advanced setting where also changed allowing a change in resolution to 1280*720 which suits the laptop.

11. A VNC connection was established using the VNC viewer allowing access to the graphical user interface of the raspberry pi.

Connecting the Raspberry pi to the internet:

1. Raspberry PI 3B has a built in Wi-Fi adapter, this adapter connected to the internet but failed to resolve default gateway, the student had to bridge the Wi-Fi connection from his laptop to the wireless router, this granted the raspberry pi internet access from its LAN Port. The following commands where inserted on the terminal of the raspberry pi to grant internet access.

- *sudo sh –c 'echo "nameserver 8.8.8.8" >> /etc/resolv.conf'*
- *sudo ip route add default via 192.168.1.1*

Step 2: Installation of raspabian stretch updates, ffmpeg and ffserver and configuring the server

To install the required software packages the following commands where typed in the raspberry pi software packages.

- *Sudo apt-get update* (Updating raspabian stretch packages)
- *Sudo apt-get install libav-tools* (Installing ffserver and 44ffmpeg)

Accessing the ffserver and configuring the server:

1. Open raspberry pi terminal.

2. Type: *sudo nano /etc/bin/ffserverconf'*

3. Configuration's set can be viewed in Appendix 4.

This configuration allows the streaming of 2 channels a prerecorded mp3 and a voice input from the microphone connected to the sound card which is then connected the usb port of the raspberry pi.

Step 3: Feeding ffserver with Stream's.

An encoder called ffmpeg was used to feed the streams to ffserver.

Feeding mp3 stream:

- Open terminal
- Change directory to the location of the mp3 : *cd /home/pi/Desktop/Music*
- Input stream using the following command: *ffmpeg –I radio.mp3* *http://localhost:8090/Music.ffm*

Feeding the Voice audio stream:

Making the sound card the default input/output sound port

- Open the /home/pi directory.
- Allow the viewing of hidden files and create a file called .asoundrc
- Type: sudo nano */proc/asound* To find out which device number is allocated to the sound card

- Insert the following

pcm.!default {
type hw
card 1#Device number
}
ctl.!default {
type hw
card 1 #Device number
}

Feeding the voice stream:

- Open terminal
- Type: *ffmpeg -re -f alsa -acodec pcm_s16le -ac 1 -ar 44100 -i hw:1,0 -af "highpass=f=200, lowpass=f=3000" -preset ultrafast -acodec mp2 -b:a 64k http://localhost:8090/live.ffm*

Automating ffserver to launch at boot:

- Created a shell script named .Autorun.sh and made it executable using the command *chmod 775 Testing Script.py*.
- Launched *crontab e* and inserted the following *@reboot/home/pi/.Autorun.sh.*

Feeding stream's to server at boot:

- Python script named *Testing Script.py* was created.
- By creating a .desktop file in /home/pi/.config/autostart/input.desktop and inserting:
  [Desktop Entry]
  Exec= sudo python /home/pi/Testing Script.py
  Type=Application
- The streams are input as soon as the raspberry pi Desktop appears.

3.7 Network Router

Huawei: Home Gateway Model: HG630 V2 created a local network between the raspberry pi's and the android devices.

<Stream stat.html>

Format status

 # Only allow local people to get the status

ACL allow localhost

ACL allow 192.168.1.5

</Stream>

The configuration above of the ffserver allows the ffserver to stream the voice and audio stream to a network. In this case the small network created by the wireless router.

To access the stream's ffserver created this URL:

*rstp://192.168.1.5:5554/*Msulive: for the voice stream.

And

rstp://192.168.1.5:554/Msump3 for the audio stream.

The streams can also be accessed via

http://192.168.1.5:8090/Msulive: for the voice stream

And

http://192.168.1.5:8090/Msumusic: for the audio stream

Not the http streams do not allow live steaming so there will be a huge delay in hearing the audio and voice. The raspberry pi are connected to router using CAT5e utp cables and the android devices using the WI-FI.

3.7 The Radio

[17] And [18] helped in the design of this radio

Main Components where used to build the streaming server, are

1. **Raspberry pi 3B and peripherals:** Which was acting is acting as the Radio. Peripherals include a 5V, 2.1Amp Samsung charger to power the raspberry pi, and a 16GB memory to provide memory to the pi
2. **16 X 2 Module Hd44780 Display Green:** Used to display current state of the radio.
3. **Momentary Contact Push button:** Used to play, stop and change channels in the radio.
4. **Huawei: Home Gateway Model: HG630 V2:** Used to provide network between pi and students PC.
5. **2*Cat 5e Universal twisted Pair:** Used to connect router to students PC and router to raspberry pi.
6. **HP Pavilion**: Used to provide peripherals to raspberry pi.
7. **Intex IT-320 Computer Multimedia Speaker**: Used to provide a sound output.

Step's in 3.5.1 where repeated in order for this raspberry pi to connect to the internet and allow the laptop to act as the raspberry pi peripheral's.

3.7.1 Interfacing the hardware

***Wiring* the raspberry pi to push buttons.**

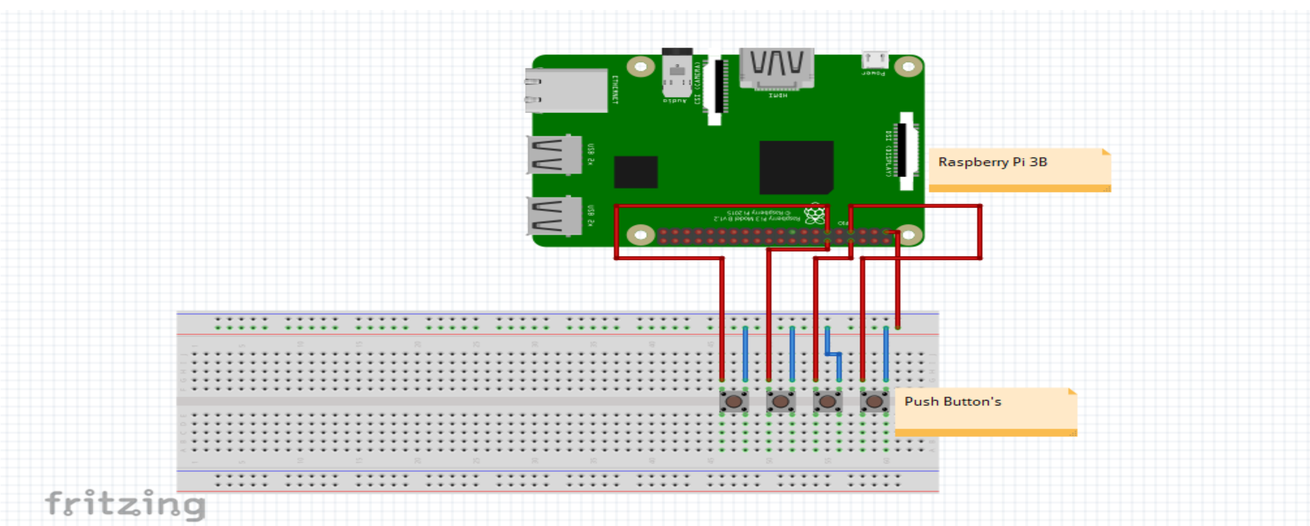


Figure 3.9: GPIO To Push Button Connections

Table 3.12: GPIO To Push Button Wiring

PIN	GPIO	FUNCTION
1	3V3	3V Power Supply To Push Button's

7	GPIO7	Play Button
8	GPIO14	Stop Button
10	GPIO15	Channel Down
11	GPIO17	Channel Up

Wiring the raspberry pi to LCD

The LCD was connected use 4 bits.

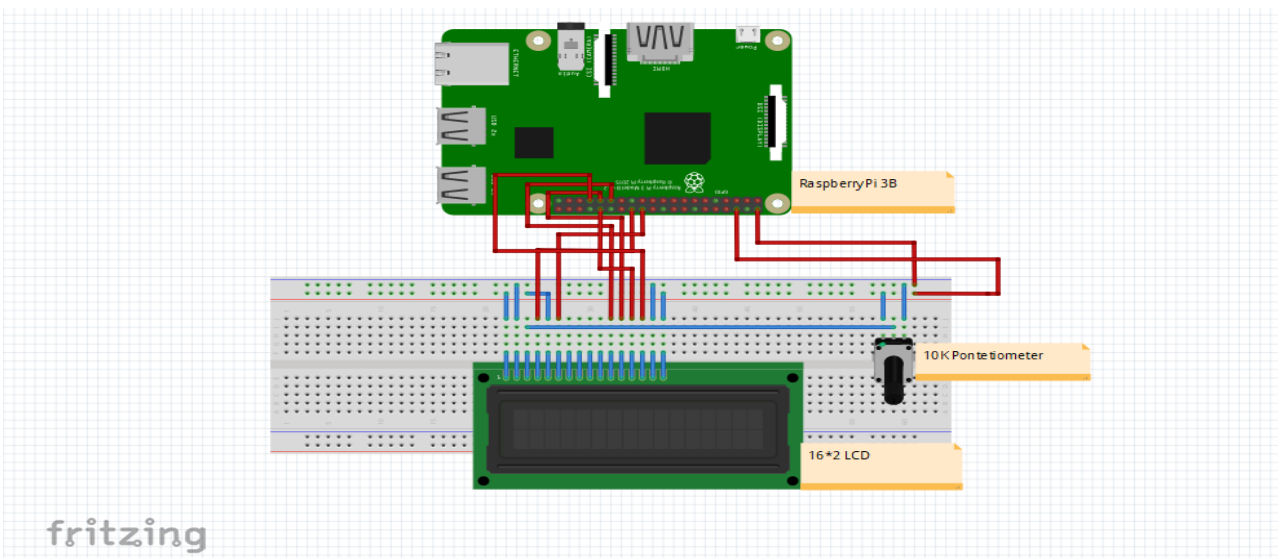


Figure 3.10: GPIO To LCD Connection's

Table 3.13: GPIO To LCD Description

LCD PIN	GPIO	GPIO PIN NUMBER	DESCRIPTION
1	n/a	6	Ground (0V)
2	n/a	2	VCC +5V
3	n/a	Note	Contrast adjustment
4	GPIO7	26	
5	n/a	6	Read/Write (RW).
6	GPIO8	24	Enable (EN)

11	GPIO5	29	Data Bit 4
12	GPIO6	31	Data Bit 5
13	GPIO12	32	Data Bit 6
14	GPIO13	2	Data Bit 7
15	n/a	2	LED Backlight Anode (5V)
16	n/a	6	LED Backlight Cathode (GND)

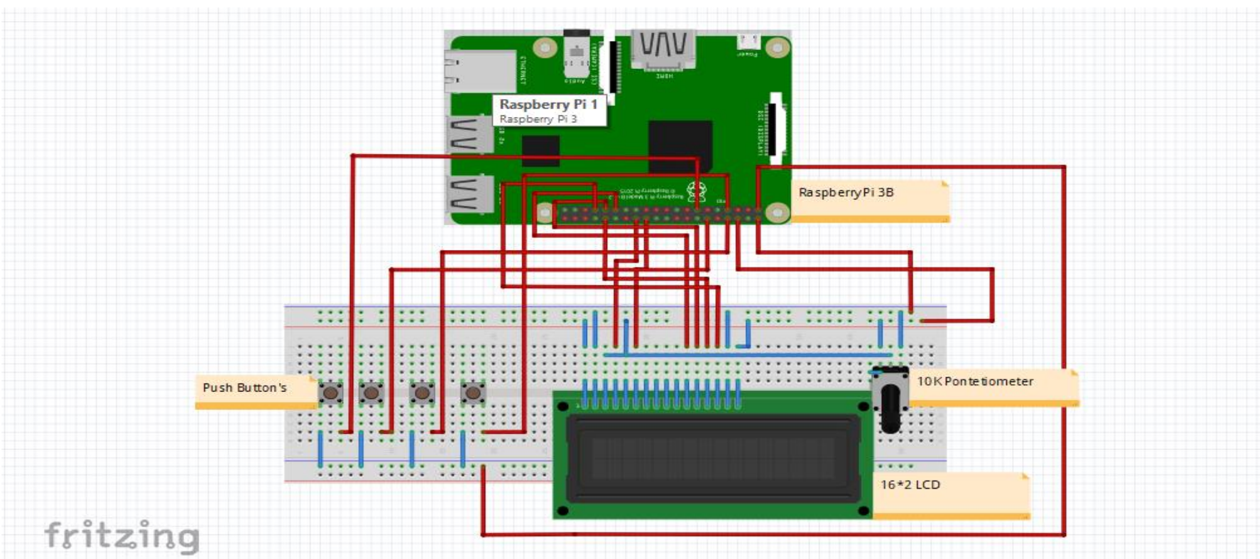


Figure 3.11: GPIO to LCD And Push Button Wiring

Intex IT-320 Computer Multimedia Speaker: used to provide an audio output for the streamed audio.

3.7.2 Software

The audio output used to create the radio was vlc media player. To download vlc media player the following command's where used.

1. *sudo apt-get update:* To update the raspabian stretch packages.
2. *sudo apt-get install vlc:* To download video LAN media player.

In order to create the radio, it had to be able to start as daemon at boot time this was accomplished by

- Creating a script called .atreboot.sh and inserting the following in the script

  #! /bin/bash

  vlc -I telnet --telnet-password=admin

- The script was made executable by typing *chmod 755 .atreboot.sh* in the terminal.
- *Crontab –e* was run and the following line was added:*@reboot /home/pi/.atreboot.*

A python code name LCD.py was modified, the main code was obtained at [19]. Displaying messages on the LCD at boot this was accomplished by:

- Creating a script called launcher.sh and inserting the following in the script
  #! /bin/bash
  cd /
  cd home/pi
  sudo python LCD.py
  cd/
- The script was made executable by typing *chmod 755 launcher.sh* and the python code by *chmod +x LCD.py* in the terminal.
- *Crontab –e* was run and the following line was added:*@reboot /home/pi/launcher*

A python code stored */home/pi/radio/bin/MSU_Radio.py* was used to create the radio application, this application was also supposed to start up as soon as the GUI appears. This was accomplished by:

- The application been made executable by typing: *chmod +x /home/pi/radio/bin/MSU_Radio.py*
- By creating a .desktop file in */home/pi/.config/autostart/myapplication*.Desktop and inserting:
  [Desktop Entry]
  Exec= sudo python /home/pi/radio/bin/MSU_RADIO.py
  Type=Application
- The application made to execute as soon as the raspberry pi GUI appears.

3.8 Android Device

A template called RTMPPlayer-master was downloaded from www.github.com and the source code was modified by the student using android studio in order to play the audio stream from the server. The audio stream can also be played by using another open source application VLC media player. Appendix 5 shows the code for the MainActivity of the android application.

REFERENCE LIST

[1] Paul Miller “https://www.theverge.com/circuitbreaker/2017/3/17/14962170/raspberry-pi-sales-12-5-million-five-years-beats-commodore-64 [Mar, 17, 2018].

[2] N.A. “https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/ [May, 23 2018].

[3] N.A. “https://en.wikipedia.org/wiki/Twisted_pair [May, 11 2018].

[4]N.A, “Category 5e UTP Solid Specifications,” datasheet, [n.d]

[5] N.A. “http://www.scpcat5e.com/bulk-cables-c-93/category-cables-c-93_106/cat5e-c-93_106_185/cat5e-utp-pvc-c-93_106_185_200/cat5egray-p-1205[n.d]

[6] XIAMEN AMOTEC DISPLAY., “Specifications of LCD Module, ADM1602K datasheet, November. 2008.

[7] N.A “https://components101.com/16x2-lcd-pinout-datasheet [May, 11 2018]

[8] Huawei Technologies Co., Ltd., “HG630 Home Gateway Product Description.HG630 Home Gateway Product Description,” Issue 203264_01 datasheet, April. 2013.

[9] N.A “https://www.adafruit.com/product/562 [May,11 2018]

[10]TTelectronics BI Technologies, “Model P160, ADM1602K datasheet, July. 2018.

[11] N.A “http://www.dx.com/p/usb-2-0-virtual-7-1-channel-sound-card-adapter-45577#.WwXIZkiFPIV[May, 11 2018]

[12] N.A “https://www.tripplite.com/virtual-7.1-channel-usb-external-sound-card~U237001 [May,11 2018]

[13] Aneex Computer Products, “Auricular Microphone with noise reduction, JY-GT008 datasheet, 2011.

[14] Intex, “Computer Multimedia Speaker, IT-320 datasheet, 2011.

[15] Yumeng Zhang, Hui Li, Jiaojiao Fu , “Audio and Video Real Time Broadcast System Based On Android Platform,” NCIS 2012, Tech. Report pp. 332–339, 2012.

[16] N.A “https://trac.ffmpeg.org/wiki/ffserver [May, 11 2018]

[17] Bob Rathbone, A definitive guide to building Internet radios using the Raspberry Pi.2018.

[18] Jack Creasey, Raspberry Pi Essentials. Birmingham: Packt Publishing, 2015.

[19] Matt “16×2 LCD Module Control Using Python:https://www.raspberrypi-spy.co.uk/2012/07/16x2-lcd-module-control-using-python/ [May, 11 2018]

Chapter 4

4 Results and Analysis

4.1 Introduction

This chapter gives the results obtained from measuring variables that constitute to Quality of Service for voice over internet protocol (VoIP). This results have been measured using an open source network monitoring software analyzer called (Wireshark). Real time transport protocol (RTP) packets where captured and analyzed to obtain useful parameters such as Bandwidth, jitter, latency, packet loss, playback period for the voice and audio channels on both wired(Raspberry pi(server) to Raspberry pi(radio)) and wireless(Raspberry pi(server) to Laptop) connection's. Results where then compared with the ITU Values for VoIP quality to give an accurate conclusion.

Note: The laptop was used instead of the android device due to Wireshark not being able to be installed on the android device.

4.2 Presentation of prototype

4.2.1 Top View

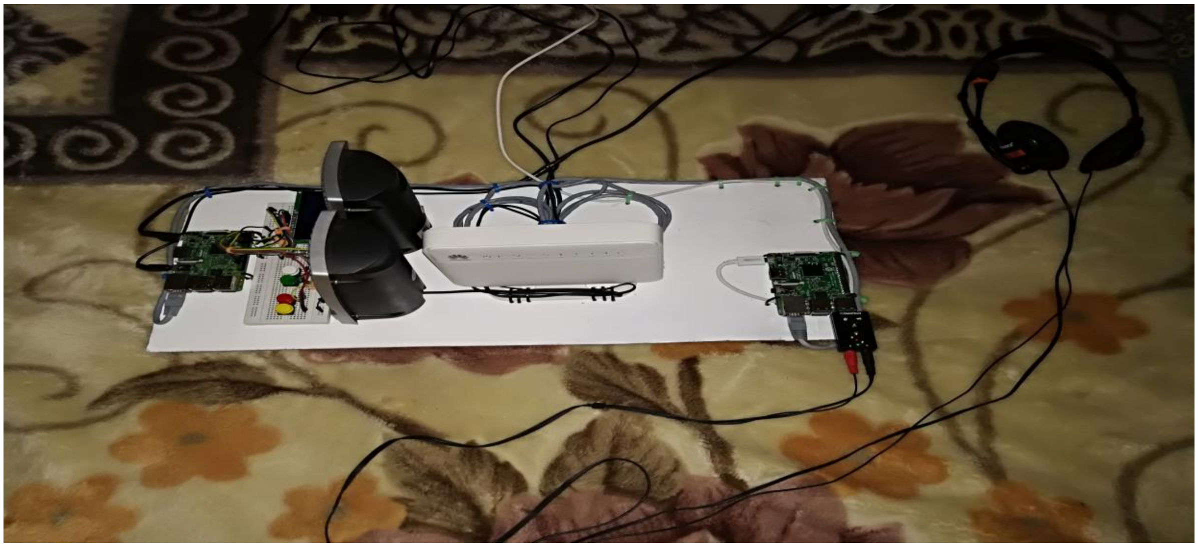


Figure 5.1: Prototype Top View

4.2.2: Front View

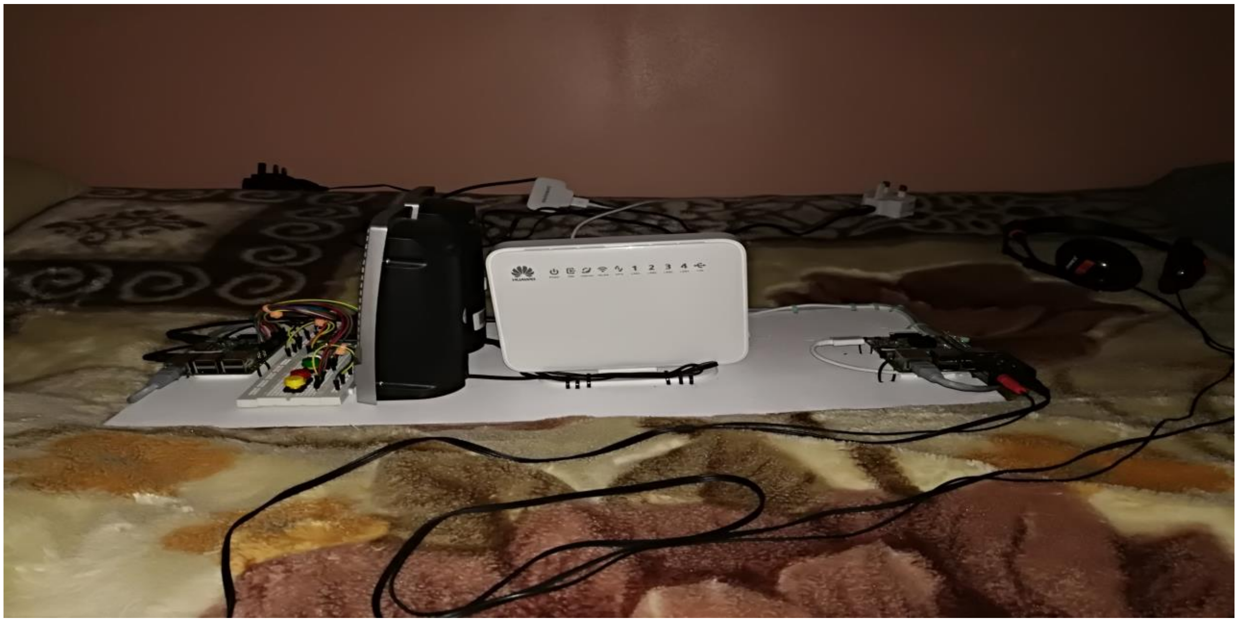


Figure 5.2: Prototype Front View

4.2.3: Back View

Figure 5.3: Prototype Back View

4.2.4 Android Application

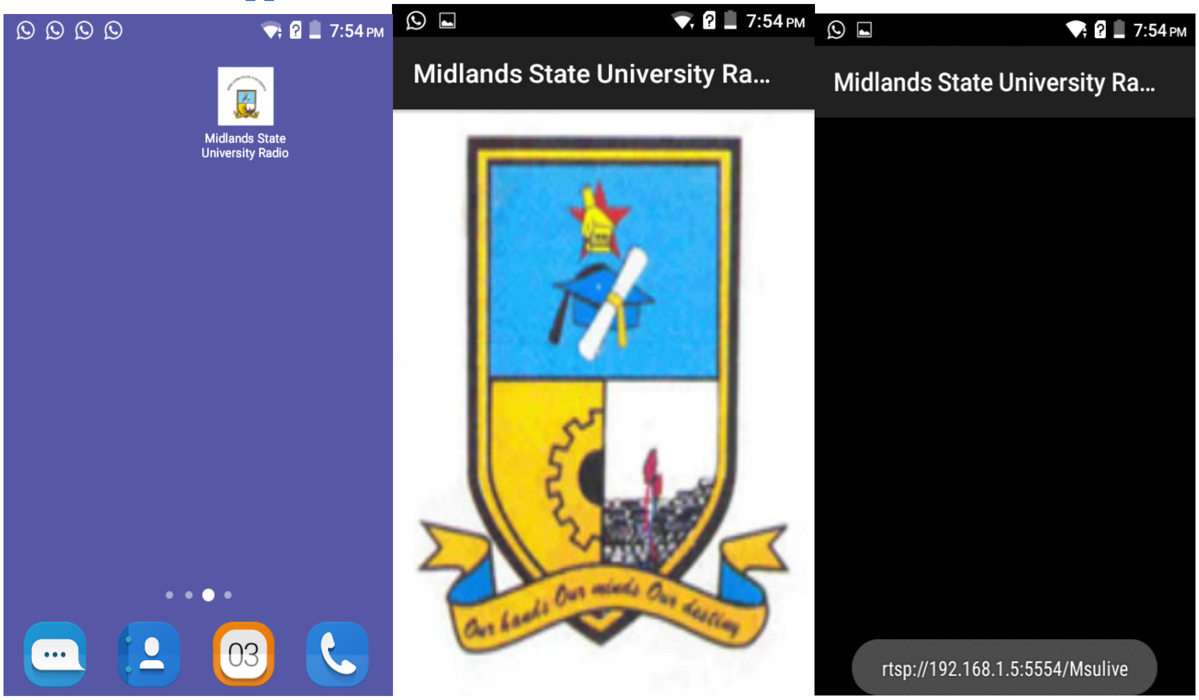


Figure 5.4: Screen Shots Of Android Application

4.3 Wireshark measurements

The experiment consisted of measurements of audio and voice streams on channels rtsp://192.168.1.5:5554/Msump3 and rtsp://192.168.1.5:5554/Msulive respectively. The measurements where taken for transmissions between Raspberry pi (server) and raspberry pi (Radio) and Raspberry pi (server) and Laptop. The results of all 4 cases have been organized in the tables and compared with the ITU values for VoIP quality with this figures obtained from [2]. The network latency value was obtained using the ping latency test and the value for the playback period was obtained by using a stop watch and average calculated.

4.3.1 Definition's

Mouth-to-ear level delays: The amount of time it took from when the word is spoke in to the mic of the transmition server and the word is herd by the listener.

Delta: [1] This is the time difference between the current and previous packet in the stream.

IP BW (kbps): [1] this parameter refers to the bandwidth consumption at the IP level that is with all headers down to layer 3.

Bandwidth: is the amount of data that can be transmitted in a fixed amount of time.

Packet loss: Voice packets fail to reach destination degrading voice quality at receiver end. Below is the packet loss equation obtained from [3]:

*Packet loss= ((Packet Sent – Packets Received)/((Packets sent))*100*

Jitter: is the variation in the latency on a packet flow between two systems, below is the jitter estimation equation obtained from [3]

$$D_{(x,y)} = (R_y - R_x)-(S_y - S_x) = (R_y - S_y) - (R_x - S_y)$$

R=Arrival time of packets

S=Rtp Time Stamps

Latency: how much time it takes for a packet of data to get from one designated point to another in a computer network, below is the Equation for latency obtained from [3]:

$$\text{Latency}=\frac{\sum(CBR\ sent\ time - CBR\ receive\ time)}{\sum CBR\ received\ packets}$$

Duration: The amount of time RTP packets where captured.

4.4 Procedure

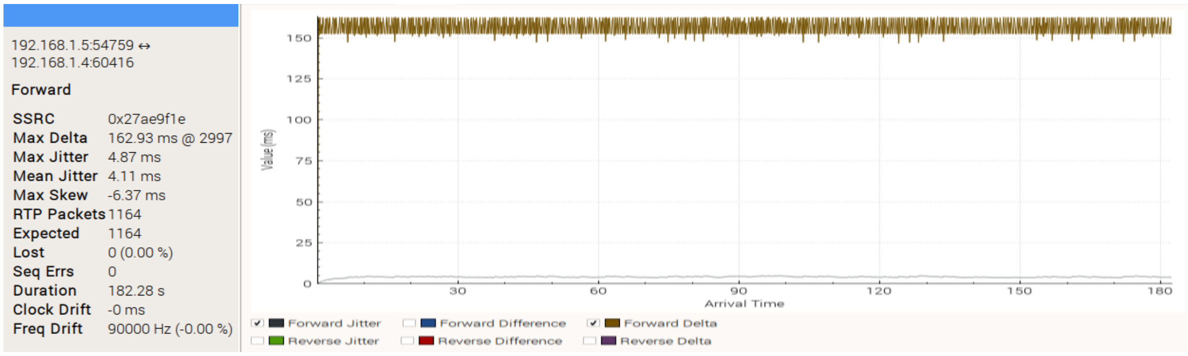
CASE 1: *Raspberry pi (Server) to Raspberry pi Radio (live voice)*

Wireshark was installed on the Raspberry pi radio, the Audio stream on channel *rtsp://192.168.1.5:5554/livemp3* was played on the radio RTP Packets where captured and the following results where obtained:

Jitter(ms)	4.11
MaxDelta (ms)	162.98
Packet loss (%)	0
IP Bandwidth(kbs)	72.68
Latency(ms)	0.874
Duration(s)	182.28

Mouth to ear delay(s)	1.82

Table 4.1: Results Of Case 1



Graph 4.1: Forward Jitter And Forward Delta For Case 1

Comparison with ITU Standards

Table 4.2: Comparison With ITU standard for Case 1

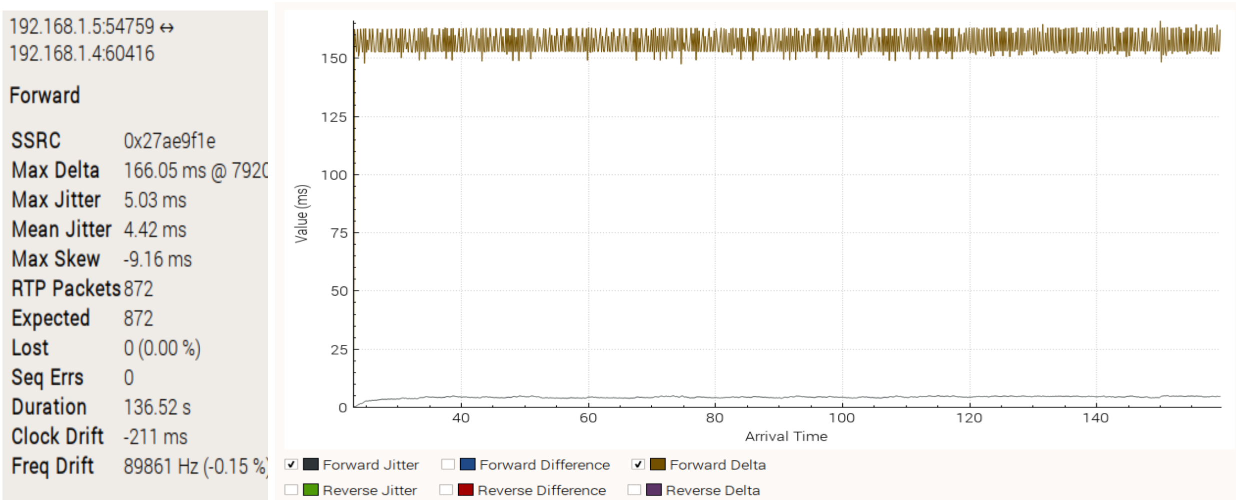
Parameter	*ITU*	*Prototype Results*
Delay(ms)	*>150ms<300ms*	*162.98*
Jitter(ms)	*<20ms*	*4.11ms*
Packet loss (%)	*<1%*	*0%*

CASE 2: *Raspberry pi (Server) to Raspberry pi Radio (Audio)*

The Audio stream on channel *rtsp://192.168.1.5:5554/livemp3* was played on the radio RTP Packets where captured and the following results where obtained:

Table 4.3: Results Of Case 2

Jitter(ms)	4.42
MaxDelta (ms)	166.05
Packet loss (%)	0
IP Bandwidth(kbs)	72.68
Latency(ms)	0.874
Duration(s)	182.28
Mouth to ear delay(s)	1.67



Graph 4.2: Forward Jitter And Forward Delta For Case 2

Comparison with ITU Standards

Table 3.4: Comparison With ITU Standard For Case 2

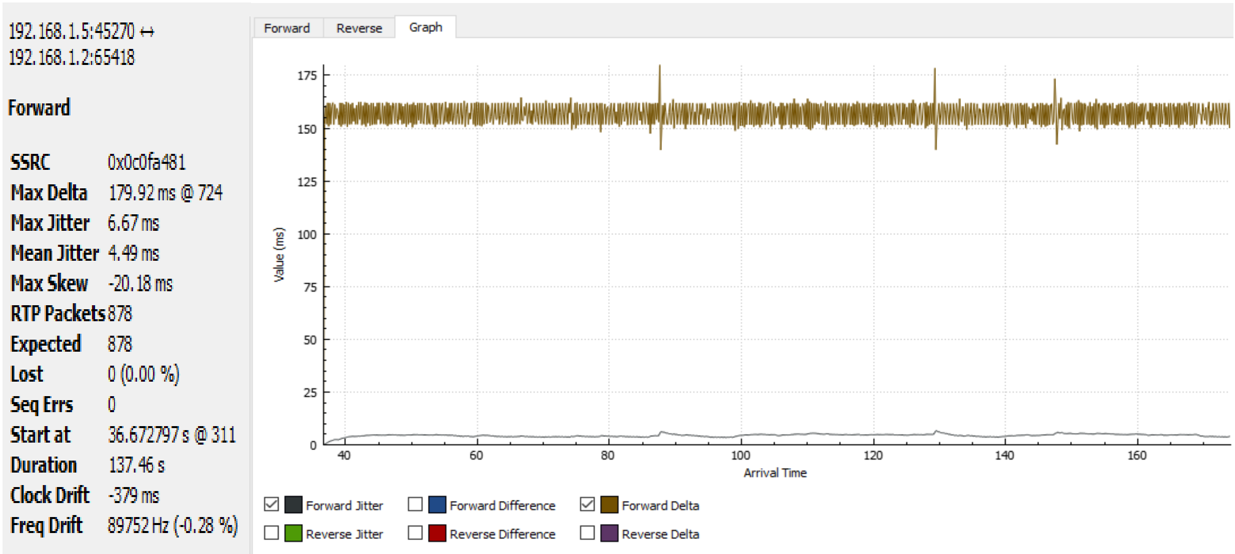
Parameter	*ITU*	*Prototype Results*
Delay(ms)	*>150ms<300ms*	*166.05*
Jitter(ms)	*<20ms*	*4.42ms*
Packet loss (%)	*<1%*	*0%*

CASE 3: *Raspberry pi (Server) to PC (Audio)*

Wireshark was installed on the PC a wireless connection was established between Server and the PC, the Audio stream on channel *rtsp://192.168.1.5:5554/Msump3* was played on the Vlc media player Packets where captured and the following results where obtained:

Table 4.5: Results Of Case 3

Jitter(ms)	4.49
MaxDelta (ms)	179.92
Packet loss (%)	0%
IP Bandwidth(kbs)	72.68
Latency(ms)	1
Duration(s)	176.51
Mouth to ear delay(s)	1.88



Graph 4.3: Forward Jitter And Forward Delta For Case 3

Comparison with ITU Standards

Table 4.6: Comparison With ITU Standard For Case 3

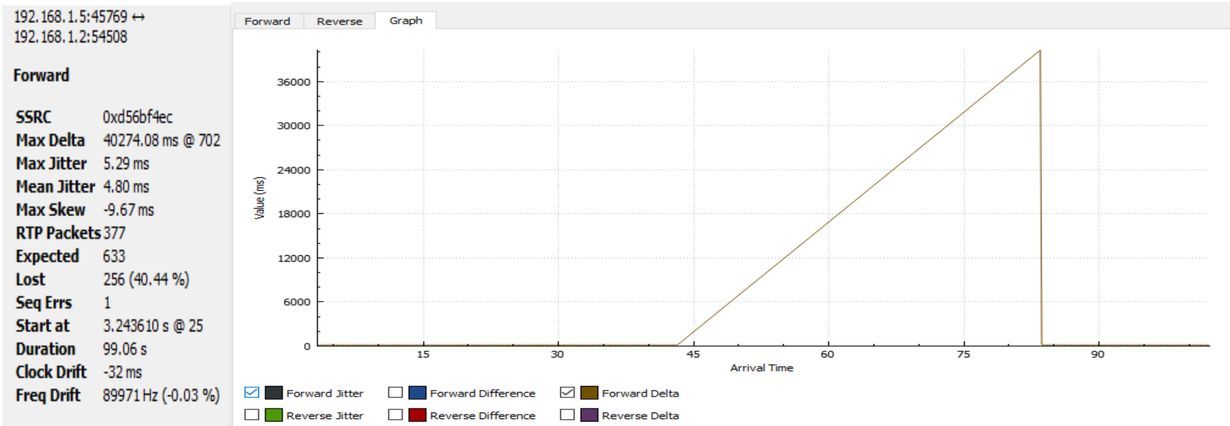
Parameter	*ITU*	*Prototype Results*
Delay(ms)	*>150ms<300ms*	*179.92*
Jitter(ms)	*<20ms*	*4.49*
Packet loss (%)	*<1%*	*0%*

CASE 4: *Raspberry pi (Server) to PC (Live voice)*

The Audio stream on channel *rtsp://192.168.1.5:5554/livemp3* was played on the radio RTP Packets where captured and the following results where obtained:

Table 4.7: Results For Case 4

Jitter(ms)	4.80
MaxDelta (ms)	40274.08
Packet loss (%)	40.44%
IP Bandwidth(kbs)	72.68
Latency(ms)	1
Duration(s)	99.06
Mouth to ear delay (s)	1.75



Graph 4.4: Forward Jitter And Forward Delta For Case 4

Comparison with ITU Standards

Table 4.8: Comparison With ITU Standard For Case 4

Parameter	*ITU*	*Prototype Results*
Delay(ms)	*>150ms<300ms*	*40274.08*
Jitter(ms)	*<20ms*	*4.80ms*
Packet loss (%)	*<1%*	*40.44%*

4.5 Conclusion

From the results it can be shown that the transmission from server to receiver complied with the ITU standard for both transmission medium. The exception was the transmission of voice from server to the receiver on the wireless medium in which the delay and packet loss was the way above the required parameter’s, though the quality of the voice was good and the mouth to ear delay’s where below 2 sec as when the stream was using the Cat 5e UTP cable as its transmission medium.

The bandwidth parameter’s where constant for all cases with the latency for wired transmission higher than that for wireless transmission. Though the system gave more positive results on the wired transmission medium than the wireless medium.

Medium	Application	Degree of symmetry	Typical data rates	Key performance parameters and target values			
				One-way delay	Delay variation	Information loss**	Other
Audio	Conversational voice	Two-way	4-64 kb/s	<150 msec preferred* <400 msec limit*	< 1 msec	< 3% packet loss ratio (PLR)	
Audio	Voice messaging	Primarily one-way	4-32 kb/s	< 1 sec for playback < 2 sec for record	< 1 msec	< 3% PLR	
Audio	High quality streaming audio	Primarily one-way	16-128 kb/s	< 10 sec	< 1 msec	< 1% PLR	
Video	Videophone	Two-way	16-384 kb/s	< 150 msec preferred <400 msec limit		< 1% PLR	Lip-synch : < 80 msec
Video	One-way	One-way	16-384 kb/s	< 10 sec		< 1% PLR	

Provisional values

Figure 4.5: ITU-T performance standard for audio and video applications [3]

REFERENCE LIST

[1] Yoram Orzach, Network Analysis Using Wireshark Cookbook. Birmingham: Packt Publishing, 2013.

[2] S. Vadivelu, "Evaluating the Quality of Service in VoIP and comparing various encoding techniques," University of Bedfordshire, Bedfordshire, 2011.

[3] Al-Begain et al, Multimedia Services and Applications in Mission Critical Communication Systems. Hershey, Pennsylvania: IGI Global, 2017.

[4] Jean-Yves Monfort, ' Basic requirements to Quality of Service (IP centric) ', Geneva France, 25 May 2003.

Chapter 5

5 Conclusion

5.1 Introduction

This chapter give's assessments that can be drawn from this project and the conclusions drawn from the demonstration of the working principle behind the intranet radio prototype.

5.2 Discussion

The prototype of the intranet radio was built and the results to test the quality of the system as it fares against Qos parameters on VoIP devices where compered in Chapter 4 of the document. With the system meeting the requirements set by the regulatory body Unfortunately this parameter's only measure Serialization delay which is defined by [1] as fixed delay that occurs when sending packets over the communication line. This delay depends on packet size and line speed. Two other delays which are handling delay ([1] delay is the time that it takes to build packets and insert voice information into them), Coding delay ([1] delay that comes from the digital processing of the voice signals) affected the system this increased the mouth to ear delay of the system to 1-2 sec.

Though the mouth to ear delay meets the one-way delay for high quality streaming server which is supposed to be less than 10sec from figure 4 it is believed that this figure can be reduced.

The audio was streamed near to some quality as the original audio. The voice was clear on the receiving end though it suffered from a lot of noise and distortion. Ffmpeg was used to try reduce the noise by applying a high pass and low pass filter though this background noise was not completely removed.

5.3 Recommendation's

- Design of a user friendly graphical user interface (For the Disk Jockey) to send a stream's with different parameters to ffserver on one channel e.g. voice and audio.
- Further research on the reduction of noise on the voice output.
- Further research has to be done how the handling delay and coding delay can be reduced in order to receive similar playback as one on a VoIP call.

- Further Work has to be done to improve the android application.

5.4 Conclusion

The Prototype was successfully designed, with the principle of operation demonstrated. From the results obtained from the prototype, with required resources this system can be implemented on a large scale and be an alternative method of radio transmission for University Campuses.

REFERENCES

[1] Yoram Orzach, Network Analysis Using Wireshark Cookbook. Birmingham: Packt Publishing, 2013.

APPENDIX 1 - Python code for intranet Radio

```
#!/usr/bin/python2.7

from Tkinter import *
import time, telnetlib
import RPi.GPIO as GPIO
import os

#global Python variables
HOST = "localhost"
PORT = "4212"
strPlaylist="/home/pi/radio/playlist"
stationarray=[]
stationtuple=()
stationcount=0
stationnum=1
SessionFlag=False
tn=telnetlib.Telnet()
userloop=True
indicator=False
debug=False

root=Tk()
root.wm_title("Midlands State University Radio")
root.config(background = "#FFFFF0")

vtime=StringVar(value="Time")
vstatus=StringVar(value="..........")
```

```
vspinnum=StringVar(value="")

def ParsePlaylist(filename):
    stationlist=[]
    printlist=[]
    global stationcount
    global stationtuple
    try:
        with open(filename, "r") as input:
            temptext=""
            for line in input:
                if ("," in line ):
                    line=line.replace("  ","")
                    line= line.split(",")
                    stationlist=stationlist + line
                    printlist=line
                    stationcount += 1
                    temptext=temptext + ( '"' + str(stationcount) + ") " + printlist[0] + '" ' )
                    stationtuple=temptext
                    if debug : print(stationcount, printlist[0])
            input.close()
    except IOError:
        print ("Unable to open ", filename)
        exit()
    return (stationlist)
    return (stationlist)

##stationarray = ParsePlaylist(strPlaylist)
```

```
def TelnetTx( outstring ):

    global tn
    global outbuf
    global inbuf
    global SessionFlag
    try:
        outbuf=outstring
        tn.write( outbuf.encode("ascii") )
        if debug : print("Write command = ", outstring)
        return (True)
    except:
        print("There was an exception with a Telnet Wr command")
        return(False)

def TelnetRx( outstring ):
    """Telnet read_until command, return inbuf=no Error"""
    global tn
    global outbuf
    global inbuf
    global SessionFlag
    try:
        outbuf=outstring
        inbuf=tn.read_until( outbuf.encode("ascii"), 1)
        if debug : print("Read until command = ", outstring)
        if debug : print("Read until return string = ", inbuf)
        return (True)
    except:
        print("There was an exception with a Telnet Rd command")
        return(False)
```

```python
def CreateSession( host, port, timeout ):
    try:
        global tn
        tn.open(host, port, timeout)
        TelnetRx("Password:")
        TelnetTx("admin\n")
        TelnetRx("> ")
        return(True)
    except:
        print("There was a problem connecting to the VLC server", host, port)
        return(False)


def ClearPlaylist():
    loopcount=5
    global outbuf
    global inbuf
    global SessionFlag
    if (SessionFlag == True):
        global tn
        TelnetTx("clear\n")
        TelnetRx("> ")
        TelnetTx("is_playing\n")
        TelnetRx("> ")
        while inbuf[0] == 49 and loopcount:
            loopcount -= 1
            if debug : print("The is_playing flag is 1")
            if debug : print("Waiting for is_playing to drop")
            TelnetTx("is_playing\n")
            TelnetRx("> ")
```

```
        if debug : print("The is_playing flag is 0")
        return(True)
    else:
        print("Session is closed")
        return(False)

def AddStationtoPlaylist( stationnum, retryflag ):

    loopcount=15
    global outbuf
    global inbuf
    global SessionFlag
    global tn
    global vstatus
    if (SessionFlag == True):
        if retryflag:
            print("Retrying....")

            vstatus.set("Retrying....")
            root.update()
        ClearPlaylist()
        if debug : print("Trying to Add station", stationnum)
        TelnetTx("add " + stationarray[ ( (stationnum * 2) - 1) ] + "\n")
        TelnetRx("> ")
        TelnetTx("is_playing\n")
        TelnetRx("> ")

        while inbuf[0] == 48 and loopcount:
            loopcount -= 1
            if debug : print("Loopcount = ", loopcount)
```

```
            if debug : print("The is_playing flag is still 0")
            if debug : print("Waiting for is_playing to set")
            TelnetTx("is_playing\n")
            if debug : print("Test is_playing flag")
            TelnetRx("> ")


        if debug : print("The is_playing flag is now 1")
        TelnetTx("is_playing\n")
        TelnetRx("> ")
        while inbuf[0] == 49 and loopcount:        #49 = ascii 1
            loopcount -= 1
            if debug : print("Wait to see if is_playing is stable, Loopcount = ", loopcount)
            time.sleep (0.15)
            TelnetTx("is_playing\n")
            TelnetRx("> ")
            #If is_playing=48=zero then we will do a retry
            if inbuf[0] == 48: AddStationtoPlaylist( stationnum, True )
            #if is_playing=49=one we assume all is ok
            if inbuf[0] == 49:
                vstatus.set("...........")
                continue
        return(True)
    else:
        print("Session is closed")
        return(False)


def stopRadio():
    global SessionFlag
```

```
    if (SessionFlag == True):
        try:
            global tn
            TelnetTx("stop\n")
            print("Trying to stop")
            TelnetRx("> ")
            return(True)
        except:
            print("There was a problem sending stop command")
            return(False)
    else:
        print("Session is closed")

def PlayRadio():
    global stationarray
    global stationcount
    global SessionFlag
    global strPlaylist
    global userloop
    if (stationcount != 0):
        SessionFlag = CreateSession(HOST, PORT, 2)
        if not SessionFlag:
            print("Unable to connect to VLC...exiting")
            exit()

        AddStationtoPlaylist( 1, False )
        print("Station 1 is the default")
        return(True)
    else:
        print("No stations to play ....exiting")
```

```
        exit()

def setupswscan():
    GPIO.setwarnings(False)
    GPIO.setmode(GPIO.BOARD)

    for i in (7, 10, 11, 12):
        GPIO.setup(i, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)
        GPIO.add_event_detect(i,GPIO.RISING,callback=button_callback)

def button_callback(channel):
    swscan()

def swscan():
    global indicator
    for i in (7, 10, 11, 12):
        #Test to see if there an event for each button
        if GPIO.event_detected(i):
            if i == 7:
                indicator=True
                playbtnPress()
            elif i == 10:
                indicator=True
                stopbtnPress()
            elif i == 11:
                indicator=True
                stationSpin.invoke("buttonup")
            elif i == 12:
                indicator=True
                stationSpin.invoke("buttondown")    #Spin down
```

```
def playbtnPress():
    if ( debug ): print ("Play Button Pressed")
    global stationnum
    AddStationtoPlaylist(stationnum, False)

def stopbtnPress():
    if  debug : print ("Stop Button Pressed")
    TelnetTx("pause\n")
    stopRadio()


def spinbtnPress():
    global stationnum
    line=vspinnum.get()
    line= line.split(")")
    if debug : print("Spin cursor on ", stationnum, line)
    stationnum=int(line[0])


def update_clock():
    vtime.set( time.strftime(" %a %B %d  %l:%M:%S %p "))
    root.after(500, update_clock)

def buildwindow():

    primaryFrame = Frame(root)
    primaryFrame.grid(row=0, column=0, sticky=W+N, padx=10, pady=10 )
    primaryFrame.grid_columnconfigure(0)
```

```
    Label(primaryFrame, text="Station").grid(row=0, column=0, padx=10, pady=10)
    statusLabel= Label(primaryFrame, textvariable=vstatus).grid(row=12, column=0, padx=10, pady=10)

    playBtn = Button(primaryFrame, text="Play", command=playbtnPress, width=10)
    playBtn.grid(row=11, column=0, sticky=W, padx=10, pady=2)

    stopBtn = Button(primaryFrame, text="Stop", command=stopbtnPress, width=10)
    stopBtn.grid(row=11, column=0, sticky=E, padx=10, pady=2)

    timeLabel= Label(primaryFrame, textvariable=vtime, font=("Helvetica", 22), fg="white", bg="black" ).grid(row=13, column=0, padx=10, pady=10)

    #Create a sub frame within primaryFrame to hold the listbox and scrollbar
    subFrame = Frame(primaryFrame )
    subFrame.grid(row=1, column=0, rowspan=10, padx=10, pady=2)

    global stationSpin
    stationSpin=Spinbox(subFrame, width=25, state="readonly", textvar=vspinnum, command=spinbtnPress, values=stationtuple )
    stationSpin.config(font=("Helvetica", 22), wrap=True, repeatdelay=1000, repeatinterval=250)
    stationSpin.pack()

stationarray = ParsePlaylist(strPlaylist)

PlayRadio()
setupswscan()
buildwindow()

update_clock()
```

root.mainloop()

APPENDIX 2 - Python code for feeding stream to ffserver

```
import os
import time

os.system("ffmpeg -i /home/pi/Desktop/Music/3.mp3 http://localhost:8090/Music.ffm")

time.sleep (5)

os.system("ffmpeg -re -f alsa -acodec pcm_s16le -ac 1 -ar 44100 -i hw:1,0 -af "highpass=f=200,
lowpass=f=3000" -preset ultrafast -acodec mp2 -b:a 64k  http://localhost:8090/live.ffm

")
```

APPENDIX 3 – Ffserver streaming Configuration's

```
HttpPort 8090
RtspPort 5554
HTTPBindAddress 0.0.0.0
MaxHTTPConnections 1000
MaxClients 1000
MaxBandwidth 1000
CustomLog -
#NoDaemon

File /tmp/live.ffm
        #FileMaxSize 5M
        #ACL allow 127.0.0.1
        ACL allow localhost
</Feed>

<Stream micmp3>
        Feed live.ffm
        Format rtp
        AudioCodec mp2
        AudioBitRate 64
        AudioChannels 1
        AudioSampleRate 44100
        NoVideo
</Stream>

<Feed Music.ffm>
 File /tmp/Music.ffm
        FileMaxSize 5M
```

```
    #ACL allow 127.0.0.1
    ACL allow localhost
</Feed>

<Stream pimp3>
    Feed Music.ffm
    Format rtp
    AudioCodec mp2
    AudioBitRate 64
    AudioChannels 2
    AudioSampleRate 44100
    AVOptionAudio flags +global_header
</Stream>

<Stream stat.html>
    Format status
    # Only allow local people to get the status
    ACL allow localhost
    ACL allow 192.168.1.5
</Stream>
```

APPENDIX 3 - Python code for feeding stream to ffserver

```
import os
import time

os.system("ffmpeg -i /home/pi/Desktop/Music/3.mp3 http://localhost:8090/Music.ffm")

time.sleep (5)

os.system("ffmpeg -re -f alsa -acodec pcm_s16le -ac 1 -ar 44100 -i hw:1,0 -af "highpass=f=200,
lowpass=f=3000" -preset ultrafast -acodec mp2 -b:a 64k  http://localhost:8090/live.ffm

")
```

APPENDIX 3 – Android Application Code

```
package com.truiton.rtmpplayer;

import android.content.Intent;
import android.content.res.Configuration;
import android.net.Uri;
import android.os.Bundle;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.util.Log;
import android.view.Gravity;
import android.view.SurfaceHolder;
import android.view.SurfaceView;
import android.view.ViewGroup.LayoutParams;
import android.widget.Toast;

import org.videolan.libvlc.IVLCVout;
import org.videolan.libvlc.LibVLC;
import org.videolan.libvlc.Media;
import org.videolan.libvlc.MediaPlayer;

import java.lang.ref.WeakReference;
import java.util.ArrayList;


public class    MainActivity extends AppCompatActivity implements IVLCVout.Callback {
    public final static String TAG = "MainActivity";
    private String mFilePath;
    private SurfaceView mSurface;
    private SurfaceHolder holder;
    private LibVLC libvlc;
    private MediaPlayer mMediaPlayer = null;
    private int mVideoWidth;
    private int mVideoHeight;


    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
```

```
        mFilePath = "rtsp://192.168.1.5:5554/Msulive";

        Log.d(TAG, "Playing: " + mFilePath);
        mSurface = (SurfaceView) findViewById(R.id.surface);
        holder = mSurface.getHolder();
    }

    @Override
    public void onConfigurationChanged(Configuration newConfig) {
        super.onConfigurationChanged(newConfig);
        setSize(mVideoWidth, mVideoHeight);
    }

    @Override
    protected void onResume() {
        super.onResume();
        createPlayer(mFilePath);
    }

    @Override
    protected void onPause() {
        super.onPause();
        releasePlayer();
    }

    @Override
    protected void onDestroy() {
        super.onDestroy();
        releasePlayer();
    }


    /**
     * Used to set size for SurfaceView
     *
     * @param width
     * @param height
     */
    private void setSize(int width, int height) {
        mVideoWidth = width;
        mVideoHeight = height;
        if (mVideoWidth * mVideoHeight <= 1)
            return;
```

```
        if (holder == null || mSurface == null)
            return;

        int w = getWindow().getDecorView().getWidth();
        int h = getWindow().getDecorView().getHeight();
        boolean isPortrait = getResources().getConfiguration().orientation ==
Configuration.ORIENTATION_PORTRAIT;
        if (w > h && isPortrait || w < h && !isPortrait) {
            int i = w;
            w = h;
            h = i;
        }

        float videoAR = (float) mVideoWidth / (float) mVideoHeight;
        float screenAR = (float) w / (float) h;

        if (screenAR < videoAR)
            h = (int) (w / videoAR);
        else
            w = (int) (h * videoAR);

        holder.setFixedSize(mVideoWidth, mVideoHeight);
        LayoutParams lp = mSurface.getLayoutParams();
        lp.width = w;
        lp.height = h;
        mSurface.setLayoutParams(lp);
        mSurface.invalidate();
    }

    /**
     * Creates MediaPlayer and plays video
     *
     * @param media
     */
    private void createPlayer(String media) {
        releasePlayer();
        try {
            if (media.length() > 0) {
                Toast toast = Toast.makeText(this, media, Toast.LENGTH_LONG);
                toast.setGravity(Gravity.BOTTOM | Gravity.CENTER_HORIZONTAL, 0,
                        0);
                toast.show();
```

```
        }

        // Create LibVLC
        // TODO: make this more robust, and sync with audio demo
        ArrayList<String> options = new ArrayList<String>();
        //options.add("--subsdec-encoding <encoding>");
        options.add("--aout=opensles");
        options.add("--audio-time-stretch"); // time stretching
        options.add("-vvv"); // verbosity
        libvlc = new LibVLC(this, options);
        holder.setKeepScreenOn(true);

        // Creating media player
        mMediaPlayer = new MediaPlayer(libvlc);
        mMediaPlayer.setEventListener(mPlayerListener);

        // Seting up video output
        final IVLCVout vout = mMediaPlayer.getVLCVout();
        vout.setVideoView(mSurface);
        //vout.setSubtitlesView(mSurfaceSubtitles);
        vout.addCallback(this);
        vout.attachViews();

        Media m = new Media(libvlc, Uri.parse(media));
        mMediaPlayer.setMedia(m);
        mMediaPlayer.play();
    } catch (Exception e) {
        Toast.makeText(this, "Error in creating player!", Toast
                .LENGTH_LONG).show();
    }
}

private void releasePlayer() {
    if (libvlc == null)
        return;
    mMediaPlayer.stop();
    final IVLCVout vout = mMediaPlayer.getVLCVout();
    vout.removeCallback(this);
    vout.detachViews();
    holder = null;
    libvlc.release();
    libvlc = null;
```

```
        mVideoWidth = 0;
        mVideoHeight = 0;
    }

    /**
     * Registering callbacks
     */
    private MediaPlayer.EventListener mPlayerListener = new MyPlayerListener(this);

    @Override
    public void onNewLayout(IVLCVout vout, int width, int height, int visibleWidth, int
visibleHeight, int sarNum, int sarDen) {
        if (width * height == 0)
            return;

        // store video size
        mVideoWidth = width;
        mVideoHeight = height;
        setSize(mVideoWidth, mVideoHeight);
    }

    @Override
    public void onSurfacesCreated(IVLCVout vout) {

    }

    @Override
    public void onSurfacesDestroyed(IVLCVout vout) {

    }

    @Override
    public void onHardwareAccelerationError(IVLCVout vlcVout) {
        Log.e(TAG, "Error with hardware acceleration");
        this.releasePlayer();
        Toast.makeText(this, "Error with hardware acceleration", Toast.LENGTH_LONG).show();
    }

    private static class MyPlayerListener implements MediaPlayer.EventListener {
        private WeakReference<MainActivity> mOwner;

        public MyPlayerListener(MainActivity owner) {
            mOwner = new WeakReference<MainActivity>(owner);
```

```
        }

        @Override
        public void onEvent(MediaPlayer.Event event) {
            MainActivity player = mOwner.get();

            switch (event.type) {
                case MediaPlayer.Event.EndReached:
                    Log.d(TAG, "MediaPlayerEndReached");
                    player.releasePlayer();
                    break;
                case MediaPlayer.Event.Playing:
                case MediaPlayer.Event.Paused:
                case MediaPlayer.Event.Stopped:
                default:
                    break;
            }
        }
    }
}
```