

© Copyright

Hussein Ahmed Fouad

Hussein Mohamed Hussein

Kareem Emad Azmy

Mohamed Ahmed Kamel

Nourhan Gamal Mohamed

Omar Mohamed Badran

7/20/2015

SkyComm Mobile access for aviation industry

By

Hussein Ahmed Fouad

Hussein Mohamed Hussein

Kareem Emad Azmy

Mohamed Ahmed Kamel

Nourhan Gamal Mohamed

Omar Mohamed Badran

Under supervision of

Dr. Hassan Mustafa Hassan

Dr. Tawfik Ismail Tawfik

A Graduation Project Report Submitted to
the Faculty of Engineering at Cairo University
in Partial Fulfillment of the Requirements for the

Degree of

Bachelor of Science

in

Electronics and Communications Engineering

Faculty of Engineering, Cairo University

Giza, Egypt

July 2016

Table of Contents

TABLE OF CONTENTS	II
LIST OF TABLES.....	VI
LIST OF FIGURES.....	VII
LIST OF SYMBOLS AND ABBREVIATIONS.....	IX
ACKNOWLEDGMENTS.....	XII
ABSTRACT	XIII
CHAPTER 1: INTRODUCTION	1
CHAPTER 2: GSM	4
2.1 INTRODUCTION	4
2.1.1 First mobile generation (1G).....	5
2.1.2 Second mobile generation (2G)	5
2.1.3 Third mobile Generation (3G)	8
2.1.4 Fourth generation network (4G)	10
2.2 GSM IN DETAILS AND WHY WE ARE USING GSM TECHNOLOGY IN OUR PROJECT?	13
2.3 GSM STRUCTURE	15
2.3.1 Base station system.....	16
2.3.2 Network switching subsystem	18
2.3.3 Operation support system	20
CHAPTER 3: SDR & OPENBTS	21
3.1 SOFTWARE DEFINED RADIO.....	21
3.1.1 Definition	21
3.1.2 SDR Block Diagram	22
3.1.3 Operation concept.....	23
3.1.4 Advantages of SDR	23
3.1.5 SDR Application.....	24
3.2 UNIVERSAL SOFTWARE RADIO PERIPHERAL (USRP).....	24
3.2.1 UHD (USRP Hardware Driver)	25
3.2.2 Products	26
3.2.3 USRP Component.....	26
3.2.4 USRP N210	27
3.2.5 Daughterboards.....	31
3.2.6 Antennas	33
3.2.7 USRP Applications	33

3.3 OPENBTS (OPEN BASE TRANSCEIVER STATION).....	35
3.3.1 OpenBTS and traditional GSM.....	35
3.3.2 OpenBTS Advantage	37
3.3.3 OpenBTS requirements.....	37
3.4 OPENBTS APPLICATION SUITE	38
3.4.1 OpenBTS	39
3.4.2 Transceiver	39
3.4.3 SMQueue	39
3.4.4 SIP router/PBX	39
3.4.5 SIPAuthServe	39
3.5 OPENBTS APPLICATION PROTOCOLS	40
CHAPTER 4: SKYCOMM SOLUTION FOR AVIATION INDUSTRY	42
4.1 MOTIVATION.....	42
4.2 SAFETY DESIGN.....	42
4.3 SOLUTION OVERVIEW.....	43
4.4 CALL FLOW DURING FLIGHT.....	44
4.4.1 Mobile Equipment	44
4.4.2 The leaky feeder.....	44
4.4.3 The ANC (Active Noise Cancellation)	48
4.4.4 Core Network.....	48
4.4.5 Satellite unit (SU)	49
4.4.6 Satellite Link.....	52
4.4.7 How to reach mobile phone from Earth station?	56
CHAPTER 5: INSTALLATION.....	59
5.1 HARDWARE COMPONENT	59
5.2 LINUX SERVER	59
5.3 SOFTWARE DEFINED RADIO.....	60
5.4 ANTENNAS	61
5.5 OPERATING SYSTEM AND DEVELOPMENT ENVIRONMENT SETUP	62
5.6 GIT COMPATIBILITY	62
5.7 DOWNLOADING THE CODE.....	63
5.8 BUILDING AND INSTALLING THE CODE.....	65
CHAPTER 6: INITIAL TESTING AND CONFIGURATION.....	72
6.1 INITIAL STATE.....	72
6.2 CONFIRM RADIO CONNECTIVITY	72
6.2.1 Ettus Research Radios	73
6.2.2 Troubleshooting Ethernet.....	76
6.3 STARTING UP THE NETWORK.....	76
6.4 THE CONFIGURATION SYSTEM AND CLI.....	77
6.4.1 Changing the Band and ARFCN.....	77
6.4.2 Ettus Research Radio Calibration	79

6.5 TESTING RADIO FREQUENCY ENVIRONMENT FACTORS	80
6.6 REDUCING NOISE	81
6.6.1 Antenna alignment	81
6.6.2 Downlink transmission power	82
6.7 STEPS TO MAKE A PHONE CALL FOR THE FIRST TIME CONFIGURATION.....	83
6.7.1 Searching for the Network	83
6.7.2 Finding the IMSI.....	84
6.7.3 Adding a Subscriber and OpenRegistration.....	85
6.7.4 Asterisk configurations	88
6.7.5 First Connection.....	91
6.7.6 Test SMS	91
6.7.7 Test Calls	93
6.8 AUTOMATIC REGISTRATION CODE.....	95
CHAPTER 7: OUTSIDE WORLD.....	98
7.1 GPRS.....	98
7.1.1 General knowledge about GPRS	98
7.1.2 Applying GPRS with OpenBTS	103
7.2 UMTS.....	109
7.2.1 General knowledge about UMTS	109
7.2.2 Why did UMTS fail and GSM work in SkyComm?.....	111
7.3 VOICE OVER IP (VOIP).....	114
7.4 TWINKLE THE SUCCEEDED TECHNIQUE	116
7.4.1 Installation Guide.....	117
7.4.2 Configuration Guide	118
CHAPTER 8: BUSINESS CASE MODEL	122
8.1 MOTIVATION.....	122
8.2 STATISTICS & RESEARCH.....	122
8.3 SYSTEM PRICING	124
8.3.1 USRP kit	124
8.3.2 RF daughterboard	125
8.3.3 Antennas	126
8.3.4 Cable.....	126
8.4 SYSTEM CAPACITY.....	127
8.4.1 The aircraft dimensions and number of passengers	127
8.4.2 Erlang B table	128
8.5 DETERMINING NUMBER OF ARFCNs (USRPs).....	129
CHAPTER 9: FEMTOCELL VS. USRP	131
9.1 FEMTOCELL DESIGN	131
9.1.1 The Femtocell Concept.....	132

9.1.2 Typical Deployment	133
9.2 FEMTOCELL OR USRP?	134
CHAPTER 10: CONCLUSION	136
REFERENCES	139
APPENDIX A: GAUSSIAN MINIMUM SHIFT KEYING (GMSK).....	140
A.1 GMSK BASICS	140
A.2 GMSK MODULATION	141
A.3 ADVANTAGES OF GMSK	142
A.4 DISADVANTAGES OF GMSK	142
APPENDIX B: OPENBTS CONFIGURATIONS	143
B.1 ALL OPENBTS PARAMETERS CONFIGURATIONS	143
APPENDIX C: ERLANG B TABLE-BLOCKED CALLS CLEARED MODEL	147

List of tables

Table 2-1: Transport technology of 2G	7
Table 2-2: Transport technology of 3G	11
Table 2-3: History of GSM.....	15
Table 3-1: Different types of USRPS	26
Table 3-2: USRP Specifications	28
Table 3-3: List of available daughterboards	32
Table 3-4: List of available Antennas	34
Table 8-1: list of available USRPs Prices.....	124
Table 8-2: list of available Daughterboard prices	125
Table 8-3: list of available antennas prices	126
Table 8-4: list of cables prices.....	126
Table 8-5: number of kits and number of subscribers with fixed Erlang	130
Table C-1: Erlang B Table	147

List of figures

Figure 2-1: Evolution of mobile & fixed subscribers.....	4
Figure 2-2: GSM network architecture.....	16
Figure 2-3: Base Station Subsystems and Network Switching System.....	16
Figure 2-4: Different BTS Cell Sizes	17
Figure 2-5: GSM Base Transceiver Station.....	18
Figure 3-1: Software Defined Radio Block diagram.....	22
Figure 3-2: USRP N210	28
Figure 3-3: USRP N210 internal configuration.....	29
Figure 3-4: RF Daughterboard Frequency Coverage	33
Figure 3-5: OpenBTS VS Traditional GSM.....	36
Figure 3-6: OpenBTS application suite	38
Figure 4-1: Design flow of SkyComm	43
Figure 4-2: Leaky Feeder construction.....	44
Figure 4-3: Elevation angel illustration.....	53
Figure 4-4: Satellite Bands	55
Figure 4-5: Uplink & downlink flow.....	57
Figure 4-6: Optical Fiber Communication	58
Figure 5-1: Rubber-duck antenna.....	61
Figure 5-2: Component architecture.....	68
Figure 6-1: Antenna alignment.....	82
Figure 6-2: Android carrier selection	83
Figure 6-3: Asterisk and debug CLI.....	89
Figure 6-4: SQLite DataBase.....	90
Figure 7-1: Overview of GPRS	100
Figure 7-2: Authentication process in 2G technology	111
Figure 7-3: Authentication vector generation.....	112
Figure 7-4: Authentication in user SIM.....	113
Figure 7-5: Startup window of Twinkle GUI	119
Figure 7-6: Twinkle user profile.....	119
Figure 7-7: Twinkle system settings.....	120
Figure 7-8: Established calls with GSM and softphone on twinkle	121
Figure 7-9: Close Twinkle correctly from star	121
Figure 8-1: Airbus A380-800	127

Figure 8-2: Airbus A380 Dimensions	128
Figure 9-1: Architecture overview of a femtocell network	133
Figure A-1: Signal using MSK modulation.....	140
Figure A-2: Spectral density of MSK and GMSK signals	141
Figure A-3: Generating GMSK using a Gaussian filter and VCO	141
Figure A-4: Block diagram of I-Q modulator used to create GMSK.....	142

List of symbols and abbreviations

ADC	Analog-to-Digital Converter
ADIRU	Air data inertial reference unit
AMPS	Advance Mobile Phone Service
AP	Access Point
ARFCN	Absolute radio frequency channel number
AUC	Authentication Center
BPU	Base Processing Unit
BSC	Base Station Controller
BSS	Base Station Subsystem
BSU	Beam-steering unit
BTS	Base Transceiver Station
DAC	Digital-to-Analog Converter
dBm	Decibel-milliwatt
DDC	Digital Down Conversion
DLNA	Duplexer Low Noise Amplifier
DUC	Digital Up Conversion
EIR	Equipment Identity Register
ETSI	European Telecommunication Standards Institute
FCS	FemtoCell convergence server
FDMA	Frequency Division Multiple Access
FEC	Frequency Error Correction
FMPA	Flange Mounted Power Amplifier
FPGA	Field-Programmable Gate Array
FSK	Frequency Shift Key
FTTH	Fiber to the home
GEO	Geostationary orbit
GigE	Gigabit Ethernet
GMSK	Gaussian Minimum Shift Keying
GPRS	General Packet Radio Service
GPS	Global Positioning System

GSM	Global Systems for Mobile Communications
HGA	High Gain Antenna systems
HLR	Home location Register
HNBAP	Home Node B Application Part
HPA	High-power amplifiers
IAX	Inter-Asterisk exchange
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IF	Intermediate frequency
IMEI	International Mobile Equipment Identity
IMT-2000	International Mobile Telecommunications-2000
IP	Internet Protocol
IS-95	Interim Standard 95
ISI	Inter Symbol Interference
ISP	Internet service provider
kbps	kilobits per second
LEO	Low earth orbit
LNA	Low-noise amplifiers
LNB	Low-noise block down converter
MEO	medium earth orbit
MMS	Multimedia message service
MSC	Mobile Switching Center
MSISDN	Mobile Station International Subscriber Directory Number
MSK	Minimum Shift Keying
NDA	Non-Disclosure Agreement
NMT-450	Nordic Mobile Telephone-450
NMT-900	Nordic Mobile Telephone-900
NSS	Network Switching Subsystem
OMC	operation and maintenance center
OSS	Operation Support System
PSK	Phase Shift Keying
PSTN	Public Switched Telephone Network
QoS	Quality of service

QPSK	Quadrature Phase Shift Keying
RF	Radio Frequency
ROF	Radio over fiber
RTP	Real-Time Transport Protocol
RUA	RANAP User Adaptation
RX	Receiver
SCDMA	Synchronous Code Division Multiple Access
SDR	Software Defined Radio
SDU	Satellite data unit
SIM	subscriber identification module
SIP	Session Initiation Protocol
SMTP	Simple Mail Transfer Protocol
TACS	Total Access Communication System
TDMA	Time Division Multiple Access
TX	Transmitter
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunications System
USRP	Universal Software Radio Peripheral
VLR	Visitor Location Register
VoIP	Voice over IP
VSAT	Very small aperture terminal
WAP	Wireless Application Protocol
W-CDMA	Wide band Code Division Multiple Access
WIMAX	Worldwide Interoperability for Microwave Access

ACKNOWLEDGMENTS

This project would not have been possible without the support of many people. Many thanks to our advisers, Dr. Hassan Mustafa and Dr. Tawfik Ismail, who helped make some sense of the concept of the project and provided all their efforts to enable us complete the project and get best results. Thanks to Cairo University and our beloved department. And finally, thanks to our families, and numerous friends who endured this long process with us, always offering support and love.

ABSTRACT

The need to be always connected through smart devices is becoming increasingly important to people, even these needs are required during flights. Since the beginnings of 2015 about 28 million passengers have connected to the network inside various planes and started to benefit from the services like calling, SMS and others that are supported by this solution. This solution has been a trend in many air flights companies such as Air France, Emirates and Etihad Airways. A survey from Air France Company stated that SMS is the most popular mobile service onboard the plane, followed by accessing social media applications. SkyComm is a solution for the previous problem which provides safe usage for mobile phones onboard.

Chapter 1

Introduction

The future of business and communications is becoming too correlated to each other, spurred by rapid innovation in the cellphone industry. "Forbes" magazine estimates that there are more than 4 billion mobile devices in use among 6.8 billion people worldwide at any time. The adaptability and portability of tablets, smartphones and text messaging devices offers an unlimited global outreach that forward-looking businesses can't afford to ignore.

One of mobile technology's most appealing aspects for businesses is the ability to instantly share or upload documents, emails and photos. Instead of being tied to an office, employees can work from any location. Managers can choose numerous applications to organize schedules more efficiently. For example, you can download files from a home computer without being there. These options greatly reduce or eliminate time spent on mundane tasks, which allows the owner to focus on running the business.

Personal mobile communication is becoming increasingly important to people, especially during flights. Since the beginnings of 2015 about 28 million passengers have connected to the network inside various planes and started to benefit from the services like calling, SMS and others that are supported by this solution. This solution has been a trend in many air flights companies such as Air France, Emirates and Etihad Airways. A survey from Air France Company stated that SMS is the most popular mobile service onboard the plane, followed by accessing social media applications. SkyComm is a solution for the previous problem which provides safe usage for mobile phones onboard. Simply this mobile access service is started on land, but during takeoff and landing the system is automatically turned off (using a code in the USRP), it turns on automatically at an altitude above 3000 meters above ground level. When the system is on, the passengers can simply switch on their devices making sure Airplane mode is off. Afterwards the customer turns on the data roaming (Note that the home operator must have a roaming agreement with the

SKYCOMM Company), and then choose manually the name of the network as an example “**SKYCOMM NETWORK**”. Once the passenger is connected, a broadcast message will be received about the pricing and the international roaming rates. There is a suggestion for ON AIR telecommunication mobile services companies that may help in having additional income by allowing other companies to have advertisement SMS’S . Finally user could keep in touch at 30,000 feet.

The main target of this project is to expand the knowledge about Soft defined radio technology, peripherals (USRP). Specifically the practical contributions offered in the Mobile access for aviation industry. We will give a complete detailed block diagram about the components and how to apply this solution in airplane. We will also highlight in this solution the replacement of the FemtoCell by the USRP kit and make a simple comparison between two solutions.

In this book we are providing a detailed solutions for mobile access in aviation industry, conceptually, the starting point for this project is to give a clear and detailed information about GSM (second generation) of mobile communications, and why we are using this technology in our project, A review of this important generation constitutes Chapters 2 and is described with engineering style and notations.

In chapter 3, the combination of OpenBTS and software defined radios are introduced in order to provide the importance of them in construction of complex radio networks purely in software.

The aim of chapter 4 is to clarify and introduce SkyComm solution inside airplane in general way which will provide call flow starting from cell phone onboard till reach destination on earth. And describe each component in the solution in details. Which means providing methodology of our project.

Getting setup and steps of installation of our system with detailed information of our configurations are described in chapter 5 and 6.

In chapter 7 we are discussing all the ways to go outside the GSM network in order to communicate with outside world.

All about marketing and expenses of our idea and how to sell it in the aviation industry, also the system capacity are described in chapter 8.

In chapter 9 we are providing basic concept of FemtoCell and how it can be used in our solution instead of USRP.

Finally in chapter 10 we are providing our project and research results with brief conclusion.

Chapter 2

GSM

2.1 Introduction

From the early analog mobile generation (1G) to the last implemented fourth generation (4G) the paradigm has changed. The new mobile generations do not pretend to improve the voice communication experience but try to give the user access to a new global communication reality. The aim is to reach communication ubiquity (every time, everywhere) and to provide users with a new set of services. The growth of the number of mobile subscribers over the last years led to a saturation of voice-oriented wireless telephony. From a number of 214 million subscribers in 1997 to 1.162 million in 2002, it is predicted that by 2010 there will be 1700 million subscribers worldwide.

(See [Figure 2-1](#)).

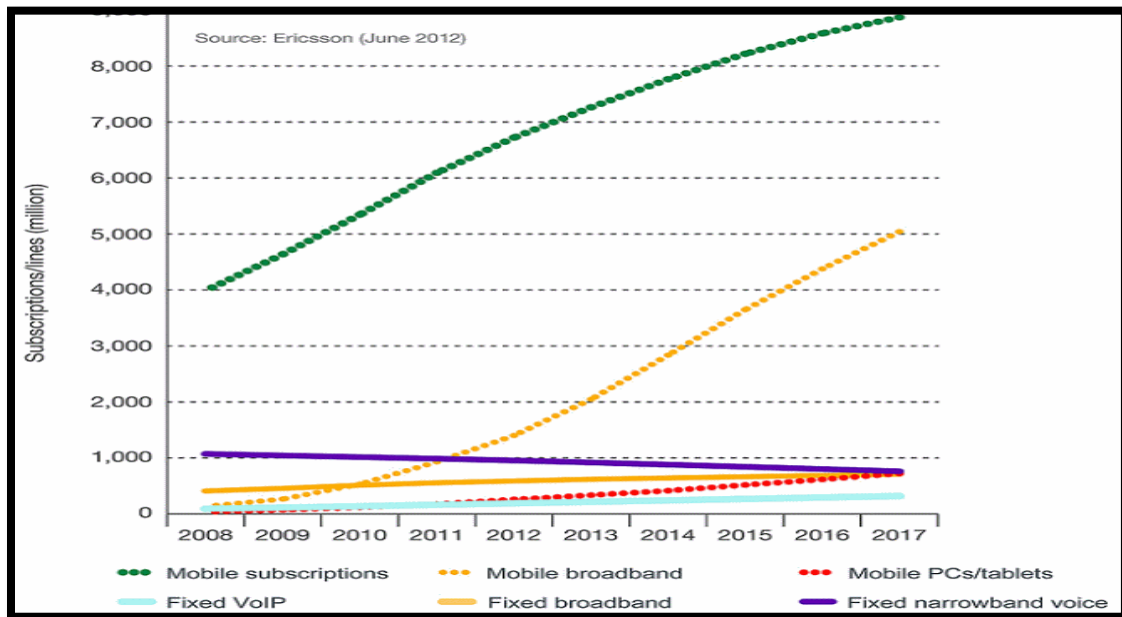


Figure 2-1: Evolution of mobile & fixed subscribers

It is now time to explore new demands and to find new ways to extend the mobile concept. The first steps have already been taken by the 2.5G, which gave users access to a data network (e.g. Internet access, MMS - Multimedia Message Service). However, users and applications demanded more communication power. As a response to this demand a new generation with new standards has been developed - 3G. In spite of the big initial euphoria that evolved this technology.

2.1.1 First mobile generation (1G)

The first operational cellular communication system was deployed in the Norway in 1981 and was followed by similar systems in the US and UK. These first generation systems - such as AMPS, TACS, NMT-450, and NMT-900- provided voice transmissions by using frequencies around 900 MHz and analogue modulation.

2.1.2 Second mobile generation (2G)

The second generation (2G) of the wireless mobile network was based on low-band digital data signaling. The most popular 2G wireless technology is known as Global Systems for Mobile Communications (GSM). The first GSM systems used a 25MHz frequency spectrum in the 900MHz band. Using FDMA (Frequency Division Multiple Access), which is a standard that lets multiple users access a group of radio frequency bands and eliminates interference of message traffic, is used to split the available 25MHz of bandwidth into 124 carrier frequencies of 200 kHz each. Each frequency is then divided using a TDMA (Time Division Multiple Access) scheme into eight time slots and allows eight simultaneous calls on the same frequency. This protocol allows large numbers of users to access one radio frequency by allocating time slots to multiple voice or data calls. TDMA breaks down data transmission, such as a phone conversation, into fragments and transmits each fragment in a short burst, assigning each fragment a time slot. With a cell phone, the caller does not detect this fragmentation.

Today, GSM systems operate in the 900MHz and 1.8 GHz bands throughout the world with the exception of the Americas where they operate in the 1.9 GHz band. Within Europe, the GSM technology made possible the seamless roaming across all countries.

The Second Generation (2G) wireless networks are based mostly on circuit switched technology, are digital and expand the range of applications to more advanced voice services. 2G wireless technologies can handle some data capabilities such as fax and short message service at the data rate of up to 9.6 kbps, but it is not suitable for web browsing and multimedia applications. So-called '2.5G' systems recently introduced enhance the data capacity of GSM and mitigate some of its limitations. These systems add packet data capability to GSM networks, and the most important technologies are GPRS (General Packet Radio Service) and WAP (Wireless Application Protocol). WAP defines how Web pages and similar data can be passed over limited bandwidth wireless channels to small screens being built into new mobile telephones. At the next lower layer, GPRS defines how to add IP support to the existing GSM infrastructure.

GPRS provides both a means to aggregate radio channels for higher data bandwidth and the additional servers required to off-load packet traffic from existing GSM circuits. It supplements today's Circuit Switched Data and Short Message Service. GPRS is not related to GPS (the Global Positioning System), a similar acronym that is often used in mobile contexts. Theoretical maximum speeds of up to 171.2 kilobits per second (kbps) are achievable with GPRS using all eight time slots at the same time. This is about ten times as fast as current Circuit Switched Data services on GSM networks. However, it should be noted that it is unlikely that a network operator will allow all time slots to be used by a single GPRS user. Additionally, the initial GPRS terminals (phones or modems) are only supporting only one to four time slots. The bandwidth available to a GPRS user will therefore be limited. All these wireless technologies are summarized in [Table 2-1](#).

Table 2-1: Transport technology of 2G

Transport technology	description	Typical use/data transmission speed	Pros/Cons
TDMA	Time Division Multiple Access is 2G technology	Voice and data Up to 9.6kbps	low battery consumption, but transmission is one way, and its speed pales next to 3G technologies
GSM	Global System for Mobile Communications is a 2G digital cell phone technology	Voice and data. This European system uses the 900MHz and 1.8GHz frequencies. In the United States it operates in the 1.9GHz PCS band up to 9.6kbps	Popular around the globe. Worldwide roaming in about 180 countries, but GSM's short messaging service (GSM-SMS) only transmits one-way, and can only deliver messages up to 160 characters long
GPRS	General Packet Radio Service is a 2.5G network that supports data packets	Data Up to 115kbps; The AT&T Wireless GPRS network transmits data at 40kbps to 60kbps	Messages not limited to 160 characters, like GSM SMS
EDGE	Enhanced Data GSM Environment is a 2G digital network	Data Up to 384kbps	May be temporary solution for operators unable to get W-CDMA licenses

2.1.3 Third mobile Generation (3G)

While GSM technology was developed in Europe, CDMA (Code Division Multiple Access) technology –which is the air interface of 3G – was developed in North America. CDMA uses spread spectrum technology to break up speech into small, digitized segments and encodes them to identify each call. CDMA distinguishes between multiple transmissions carried simultaneously on a single wireless signal. It carries the transmissions on that signal, freeing network room for the wireless carrier and providing interference-free calls for the user. Several versions of the standard are still under development. CDMA promises to open up network capacity for wireless carriers and improve the quality of wireless messages and users' access to the wireless airwaves. Whereas CDMA breaks down calls on a signal by codes, TDMA breaks them down by time. The result in both cases is an increased network capacity for the wireless carrier and a lack of interference for the caller.

CDMA technology are recognized as providing clearer voice quality with less background noise, fewer dropped calls, enhanced security, greater reliability and greater network capacity.

All 2G wireless systems are voice-centric. GSM includes short message service (SMS), enabling text messages of up to 160 characters to be sent, received and viewed on the handset. Most 2G systems also support some data over their voice paths, but at painfully slow speeds usually 9.6 Kb/s or 14.4 Kb/s. So in the world of 2G, voice remains king while data is already dominant in wire line communications. And, fixed or wireless, all are affected by the rapid growth of the Internet. Planning for 3G started in the 1980s. Initial plans focused on multimedia applications such as videoconferencing for mobile phones. When it became clear that the real killer application was the Internet, 3G thinking had to evolve. As personal wireless handsets become more common than fixed telephones, it is clear that personal wireless Internet access will follow and users will want broadband Internet access wherever they go. Today's 3G specifications call for 144 Kb/s while the user is on the move in an automobile or train, 384 Kb/s for pedestrians, and ups to 2 Mb/s for stationary users. That is a big step up from 2G bandwidth using 8 to 13 Kb/s per channel to transport speech signals.

The second key issue for 3G wireless is that users will want to roam worldwide and stay connected. Today, GSM leads in global roaming. Because of the pervasiveness of GSM, users can get comprehensive coverage in Europe, parts of Asia and some U.S. coverage. A key goal of 3G is to make this roaming capacity universal.

The third issue for 3G systems is capacity. As wireless usage continues to expand, existing systems are reaching limits. Cells can be made smaller, permitting frequency reuse, but only to a point. The next step is new technology and new bandwidth.

Telecommunication Union name for 3G and is an initiative intended to provide wireless access to global telecommunication infrastructure through both satellite and terrestrial systems, serving fixed and mobile phone users via both public and private telephone networks. GSM proponents put forward the universal mobile telecommunications system (UMTS), an evolution of GSM, as the road to IMT-2000. Alternate schemes have come from the U.S., Japan and Korea. Each scheme typically involves multiple radio transmission techniques in order to handle evolution from 2G. Agreeing on frequency bands for IMT-2000 has been more difficult and the consensus included five different radio standards and three widely different frequency bands. They are now all part of IMT-2000. To roam anywhere in this "unified" 3G system, users will likely need a quintuple-mode phone able to operate in an 800/900 MHz band, a 1.7 to 1.9 GHz band and a 2.5 to 2.69 GHz band.

UMTS use the radio technology called W-CDMA (Wide band Code Division Multiple Access). W-CDMA is characterized by the use of a wider band than CDMA. W-CDMA has additional advantages of high transfer rate, and increased system capacity and communication quality by statistical multiplexing. W-CDMA utilizes efficiently the radio spectrum to provide a maximum data rate of 2 Mbps. With the advent of mobile Internet access, suddenly the circuit-based backhaul network from the base station and back has to significantly change. 3G systems are IP-centric and will justify an all-IP infrastructure. There will be no flip to 3G, but rather an evolution and, because of the practical need to re-use the existing infrastructure and to take advantage of new frequency bands as they become available, that evolution will look a bit different depending on where you are. The very definition of 3G is now an umbrella, not a single standard, however, the industry is

moving in the right direction towards a worldwide, converged, network. Meanwhile, ever-improving DSPs will allow multi-mode, multi-band telephones that solve the problem of diverse radio interfaces and numerous frequency bands. When one handset provides voice and data anywhere in the world, it will be 3G no matter what is running behind the scenes. The previous technologies are illustrated on [table 2-2](#).

2.1.4 Fourth generation network (4G)

The objective of the 3G was to develop a new protocol and new technologies to further enhance the mobile experience. In contrast, the new 4G framework try to accomplish new levels of user experience and multi-service capacity by integrating all the mobile technologies that exist (e.g. GSM - Global System for Mobile Communications, GPRS - General Packet Radio Service, IMT-2000 - International Mobile Communications, Wi-Fi - Wireless Fidelity, and Bluetooth). In spite of different approaches, each resulting from different visions of the future platform currently under investigation, the main objectives of 4G networks can be stated in the following properties:

- 1) Ubiquity
- 2) Multi-service platform
- 3) Low bit cost

Table 1-2: Transport technology of 3G

Transport technology	description	Typical use/data transmission speed	Pros/Cons
CDMA	Code Division Multiple Access is a 2G technology developed by Qualcomm that is transitioning to 3G		Although behind TDMA in number of subscribers, this fast-growing technology has more capacity than TDMA
WCDMA (UMTS)	Wide band CDMA (also known as Universal Mobile Telecommunications System-UMTS) is 3G technology a benchmark for royalty rates	Voice and data. UMTS is being designed to offer speeds of at least 144kbps to users in fast-moving vehicles Up to 2Mbps initially. Up to 10Mbps by 2005, according to designers	Likely to be dominant outside the United States, and therefore good for roaming globally. Commitments from U.S. operators are currently lacking, though AT&T Wireless performed UMTS tests in 2002. Primarily to be implemented in Asia-Pacific region
CDMA 2000 1XRTT	A 3G technology, 1xRTT is the first phase of CDMA2000	Voice and data Up to 144kbps	Proponents say migration from TDMA is simpler with CDMA2000 than W-CDMA, and that spectrum use is more efficient.

Ubiquity means that this new mobile networks must be available to the user, anytime, anywhere. To accomplish this objective services and technologies must be standardized in a worldwide scale. Furthermore the services to be implemented should be available not only to humans as have been the rule in previous systems, but also to everything that needs to communicate. In this new world we can find transmitters in our phone to enable voice and data communications (e.g. high bandwidth Internet access, multimedia transmissions), in our wrist, to monitor our vital signs, in the packages we send, so that we always know their location, in cars, to always have their location and receive alerts about an accident, in remote monitor/control devices, in animals to track their state or location, or even in plants. Based on this view, NTT DoCoMo, that has already a wide base of 3G mobile users, estimates the number of mobile communication terminals to grow in Japan from the actual 82.2 million to more than 500 million units by 2010.

Multi-service platform is an essential property of the new mobile generation, not only because it is the main reason for user transition, but also because it will give telecommunication operators access to new levels of traffic. Voice will lose its weight in the overall user bill with the raise of more and more data services.

Low-bit cost is an essential requirement in a scenario where high volumes of data are being transmitted over the mobile network. With the actual price per bit, the market for the new high demanding applications, which transmit high volumes of data (e.g. video), is not possible to be established. According to cost per bit should be between 1/10 and 1/100 of 3G systems. To achieve the proposed goals, a very flexible network that aggregates various radio access technologies, must be created. This network must provide high bandwidth, from 50-100 Mbps for high mobility users, to 1Gbps for low mobility users, technologies that permit fast hand-offs an efficient delivery system over the different wireless technologies available, a method of choosing the wireless access from the available ones. Also necessary is a QoS framework that enables fair and efficient medium sharing among users with different QoS requirements, supporting the different priorities of the services to be deployed. The core of this network should be based in Internet Protocol version 6 – IPv6, the probable convergence platform of future services (IPv4 does not provide a

suitable number of Internet addresses). The network should also offer sufficient reliability by implementing a fault-tolerant architecture and failure recovering protocols.

2.2 GSM in details and why we are using GSM technology in our project?

GSM stands for Global System for Mobile Communications originally stands for (Groupe Spéciale Mobile). It is a European standard for the mobile telecommunications and it is considered as one of the most popular standards worldwide. It was developed by the European Telecommunication Standards Institute (ETSI) with the sole purpose of describing protocols for the second generation (2G) digital cellular networks. GSM originated in 1982 and the actual commercial launch date was 1991 in Europe. In the beginning of 1994, it was approximated that 1.3 million subscribers worldwide (nearly 80 percent) uses GSM technology. It was adopted by more than 400 operators in 173 countries. The cost of making GSM calls is reducing over time due to rapid development .GSM operates in the 900 MHZ band , having an up-link band that ranges from 890 to 915 MHZ and down-link band that ranges from 935 to 960 MHZ. The two bands are separated by 45 MHZ. GSM also works in other frequency bands around 1800 and 1900 MHZ known as DCS (Digital Cellular System bands). **Table 2-3** shows the change of events in GSM history throughout the years. Moreover it uses GMSK as a modulation technique for more details about GMSK you may refer to **appendix A**.

Why we are using GSM technology in our project?

GSM is a good choice precisely although it is old. Everyone knows it works and 80% of the world's carriers are still using it. It's a proven technology that is well-suited to the target application. The specification is publicly available and in a few more years most of the essential patents will expire.

CDMA physical layers are too complex for an inexpensive all-software radio and do not scale well for low-capacity cells. CDMA capacity comes in increments of 50 or more subscriber lines and the lowest layers of your radio must process all of that bandwidth whether you intend to use it or not. By contrast, GSM capacity comes in increments of 7-8 lines and a well-managed radio can even ignore inactive parts of the signal. Beyond the technical issues, IS-95-style CDMA (including CDMA2000) is tightly controlled intellectual property. You can't even get a copy of the specification without signing a NDA and paying several hundred dollars.

Our main target is to have all functions of BTS, BSC and MSC collapsed in the OpenBTS box. Also one needs to make the OpenBTS able to connect to another OpenBTS. The conventional way of the OpenBTS project to do this is by operating each BTS as an access point to the IP-Network, with a GSM Um interface to connect.

Table 2-3: History of GSM

Years	Events
1982	CEPT establishes a GSM group in order to develop the standards for a pan-European cellular mobile system
1985	A list of recommendations to be generated by group is accepted
1987	TDMA (Time Division Multiple Access)) over FDMA (Frequency Division Multiple Access) was used. The initial MOU (Memorandum Of Understanding) is signed by telecommunication operators representing 12 countries.
1989	The GSM specification was passed to ETSI to publish it as a standard
1990	Phase 1 of the GSM specifications is delivered
1991	Commercial launch of the GSM service occurs. The DCS1800 specifications are finalized
1992	The addition of the countries that signed the GSM memorandum of understanding took place. Coverage was spread around large cities and airports
1993	Coverage of main roads GSM services started outside Europe
1994	Data transmission capabilities launched. The number of networks increased to 69 in 43 countries by the end of 1994
1996	June: 133 network in 81 country
1999	Wireless Application Protocol was launched over 130 countries with 260 million subscribers
2000	GPRS (General Packet Radio Service) came into existence

2.3 GSM Structure

The basic GSM network structure consists of four subsystems:

- 1) BSS (Base Station Subsystem)
- 2) Networking and Switching Subsystem
- 3) OSS (Operations Support System),
- 4) GPRS Core Network; which is an optional part that allows packet-based Internet connections.

Figure 2-2 shows GSM architecture with four main subsystems. While, Figure 2-3 displays the Base Station Subsystem and Network Switching System.

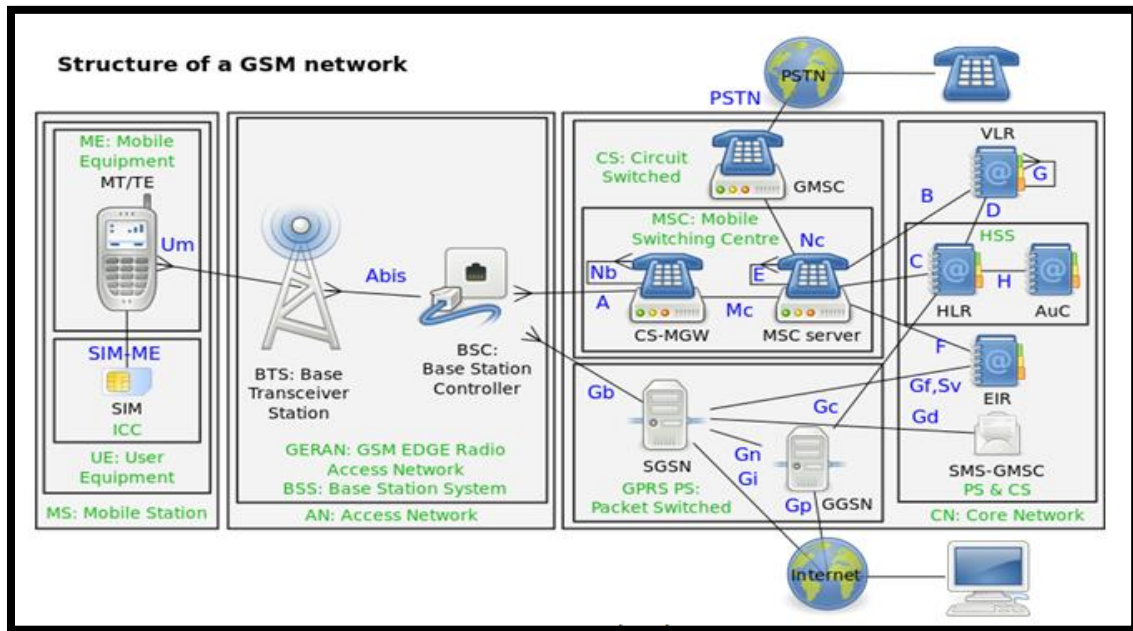


Figure 2-2: GSM network architecture

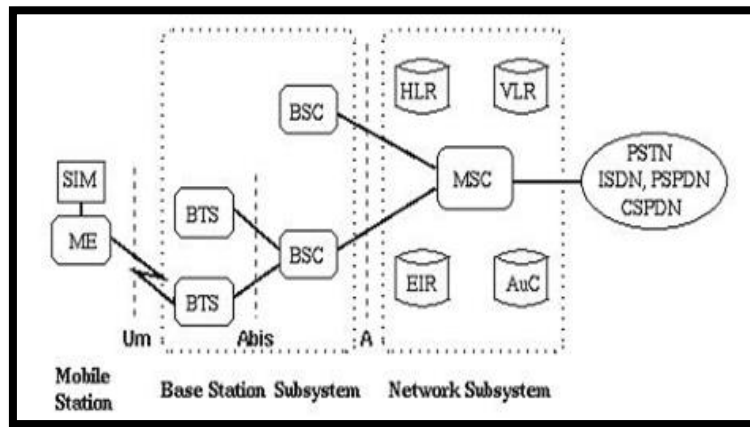


Figure 2-3: Base Station Subsystems and Network Switching System

2.3.1 Base station system

The BSS (Base station subsystem) is responsible for traffic handling and signaling between the mobile device, and Networking and Switching Subsystem. It is also responsible for managing the radio network which includes operations such as allocation of radio channels,

transmission and reception over the air interface, and many other functions. The Base Station Subsystem is composed of two main components BTS (Base Transceiver Station) and BSC (Base Station Controller).

Base transceiver station

The BTS is equipped with transceiver (antennas) which are used to transmit and receive radio signals. The BTS defines the size of the coverage of the cell according to its transmitted power and it is always found in the center of the cell. There are several cell sizes they are macro, micro, pico, femto and umbrella cells. **Figure 2.4** shows different cell sizes. Macro cells provide radio coverage served by a high power cellular base station (tower). Generally, macro-cells provide coverage larger than micro-cells. Micro cells are installed and placed on rooftops and are mainly used in cities and urban areas. Pico cells have a small radius and cover several meters; they are used mainly inside buildings. Femto-cells is typically designed for use in a home or small business. Umbrella cells are used to fill blind spots between cells. There are several important functions of the BTS, they establish and maintain connections between the MS (Mobile Station). The main function of the BTS is transmission and reception of radio signals. Moreover, it records signal strength measurements and forwards it to the BSC. Some BTS are capable of Intra-cell handover. **Figure 2-5** shows a diagram of GSM base transceiver station.

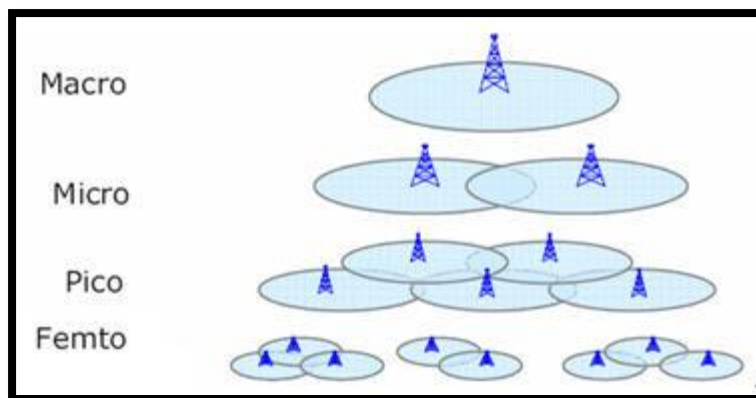


Figure 2-4: Different BTS Cell Sizes

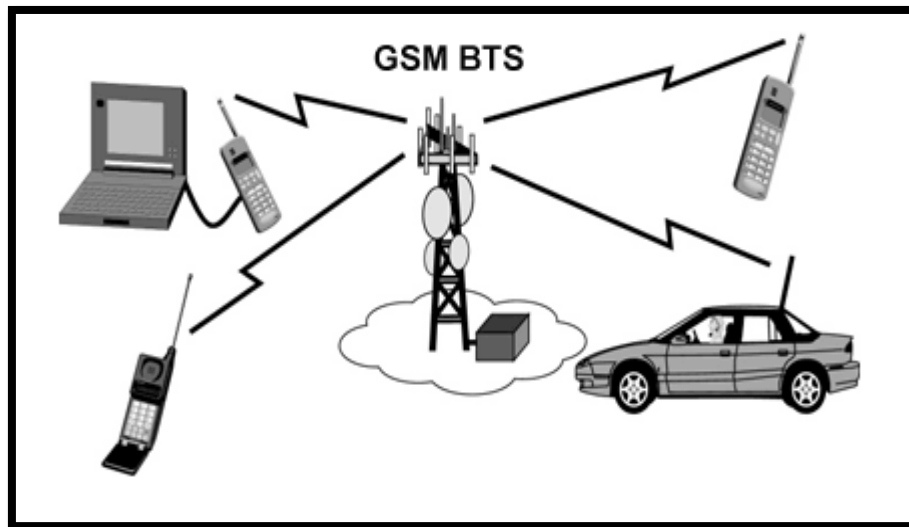


Figure 2-5: GSM Base Transceiver Station

Base station controller

The BSC acts as the brain behind the BTS. Each BSC has several BTS under its control. It is also responsible for allocation of radio channels, controls handover of calls from BTS, and supervises the performance of each BTS. In addition, it acts as a concentrator which means that lots of connections to the BTS which is low utilized which is reduced to smaller number of connections towards the MSC (Mobile Switching Center) which is highly utilized. The process in which the number of connections decreases as you go up the hierarchy means that the system becomes more organized and feasible. Plus,

2.3.2 Network switching subsystem

The NSS is the second main subsystem of the GSM network architecture and is a critical part as it carries out all the calls switching, routing and mobility management such as authentication. It includes several functional units, Mobile Switching Center, Home Location Register, Authentication Center, Visitor Location Register, and Equipment Identity Register.

Mobile switching center

This unit plays a central role in the GSM network. The MSC has controls and administration functions on several BSC. It is responsible for routing voice calls and SMS as well as taking care of charging and real time prepaid account details. Moreover it establish the end to end connection, also handles the mobility and handover requirements during a call. It also act as a gateway to other connections is provided by the GMSC (Gate way Mobile switching center). The MSC is connected to VLR, AUC, HLR and EIR.

Home location register

This is the central database used for storage and management of subscriptions. The HLR stores details about every SIM card. Each SIM has a unique identifier called IMSI (International Mobile Subscriber Identity) which is the primary key to each HLR record. It also stores MSISDN which is basically the telephone number if the SIM card which allows phone calls to be executed. It stores data like location information, and activity status.

Visitor location register

The VLR is another database that contains the subscribers that are available on the MSC that it serves. It stores a copy of the HLR profile for all the currently registered subscribers who are covered by the cells belonging to the MSC coverage area. Whenever an MSC detects a new MS in its network, in addition to creating a new record in the VLR, It also updates the HLR of the mobile subscriber new location. It stores data such as IMSI, authentication data (Black, Grey, and white list), MSISDN, HLR address of the subscriber.

Authentication center

Authentication of each SIM card to allow the connection to the GSM network is performed by the AUC. The AUC is a highly secured database that stores the secret key which is stored in each subscriber SIM card. This tries to minimize the fraud and illegal access to the network.

Equipment identity register

The EIR contains all the valid mobile equipment on the network which is authenticated using IMEI (International Mobile Equipment Identity).EIR main purpose is to track down the stolen mobile phones. The IMEI is marked as invalid if it has been reported stolen or not approved and it enters the black list.

2.3.3 Operation support system

OSS is a computer system that is connected to all the equipment in the Switching System and Base Station Center. It is the functional entity from which the network operator maintains and monitors the system. The purpose of OSS is to offer the customer cost-effective support for centralized, regional and local operational and maintenance activities that are required for a GSM network. An important function of OSS is to provide a network overview and support the maintenance activities of different operation and maintenance organizations; it provides security management, operation and performance management

After presenting an overview of GSM. All GSM stack can be written in a code this code can be implemented on the SDR technology by using what's called USRP kit. Which will be interpreted the next chapter.

Chapter 3

SDR & OpenBTS

3.1 Software Defined Radio

3.1.1 Definition

Over the last decade as semiconductor technology has improved both in terms of performance, capability and cost, new radio technologies have emerged from military, research and development labs and become mainstream technologies. One of these technologies is software defined radio.

A number of definitions can be found to describe Software Defined Radio, also known as Software Radio or SDR. The SDR Forum, working in collaboration with the Institute of Electrical and Electronic Engineers (IEEE) P1900.1 group, has worked to establish a definition of SDR that provides consistency and a clear overview of the technology and its associated benefits. Simply put Software Defined Radio is defined as "**Radio in which some or all of the physical layer functions are software defined**". In other words, Software Defined Radio (SDR) is a radio communication technology that is based on Software defined wireless communication protocols instead of hardwired implementations.

The basic principle of SDR is the reduction of the hardware dedicated to signal processing parts and its transformation into software that could be executed on a computer that is able to run such software. Frequency band, air interface protocol and functionality can be upgraded with software download and update instead of a complete hardware replacement.

3.1.2 SDR Block Diagram

A universal SDR structure with the specific software (GNU Radio) and hardware (USRP) is given in [Figure 3-1](#).

The Software-Defined Radio (SDR) structure is divided into three blocks. The left one builds the RF frontend of the hardware which serves as interface to the analog RF domain. In the second block, the intelligence of the hardware part is implemented, forming the interface between the digital and the analog world. In the third block, the whole signal processing is done - fully designed in software.

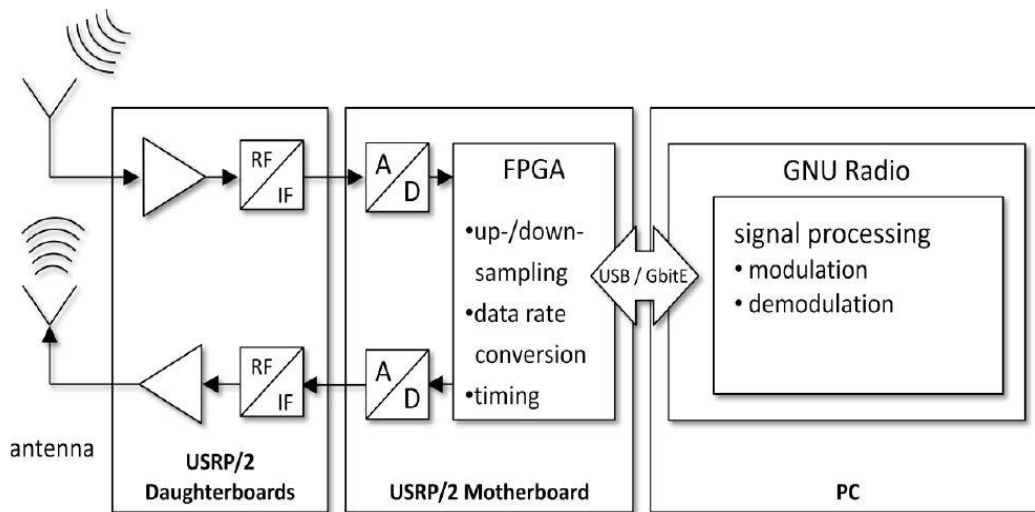


Figure 3-1: Software Defined Radio Block diagram

The interface to the analog world is given as mentioned on left side of Figure 3. An analog RF signal can be received or transmitted over antennas. The upper path (arrow towards the daughterboard) marks the receive path (Rx), the lower path describes the transmit path (TX). Both paths can operate autonomously. Daughter boards form the RF frontend of USRP and are connected to the USRP motherboard.

On USRP motherboard, the analog signals are converted to digital samples and mixed down to baseband within the FPGA. Also a decimation of the sample rate is performed. Data sampled by the FPGA are sent to the host by USB or Gigabit Ethernet respectively what is used – USRP connected to the host computer (right block in Figure 3), PC type

depends on the type of application employed, for example, if the radio is used for narrow band purposes, a regular Pentium PC should have more than enough capacity to meet the requirements but if one's implementing a radio that uses up a much bigger frequency band, one might need a more powerful PC in order to process all the data that we are using in the required time.

The GNU Radio framework controls the further signal processing capabilities. GNU Radio is an open source framework, providing various pre-assembled signal processing blocks for waveform creation and analysis in software radio development.

3.1.3 Operation concept

The ideal receiver scheme would be to attach an analog-to-digital converter to an antenna. A digital signal processor would read the converter, and then its software would transform the stream of data from the converter to any other form the application requires.

An ideal transmitter would be similar. A digital signal processor would generate a stream of numbers. These would be sent to a digital-to-analog converter connected to a radio antenna.

The ideal scheme is not completely realizable due to the actual limits of the technology. The main problem in both directions is the difficulty of conversion between the digital and the analog domains at a high enough rate and a high enough accuracy at the same time.

3.1.4 Advantages of SDR

SDR has expanded the idea of open-source and enabled amateur radio users and students to try and join the world of communications with very reasonable costs and without the need of complicated hardware, all what is needed is a Computer, a single transceiver and a software code that can be easily implemented or can be obtained from the internet, All this Software enabled the prototyping to be faster and cheaper than hardware prototyping.

SDR has the ability to receive and transmit various modulation methods using the same set of hardware. The ability to alter functionality by downloading and running new software as well as the possibility of adaptively choosing an operating frequency and a mode best

suited for prevailing conditions. In other word SDR solves the two main challenges for a wireless system, which are compatibility and spectrum usage.

From the Vendors point of view, SDR enables the implementation of a family of radio products using a common platform architecture allowing the prototyping and so faster introduction of new products and the development costs will be dramatically low. Also the use of SDR would allow bug fixing over the air or other remote reprogramming thus reducing both time and cost associated with operation and maintenance.

While for Operators, New features and capabilities could be added without requiring major modifications to the hardware as the old hardware could be used with simple modifications to the software to upgrade the whole system to work with the new features and services significantly reducing logistical support and operating expenditures.

A Software Defined Radio can easily be many different kinds of radio, often several different types at once. SDR has the potential to be a revolutionary technology that will dramatically impact the wireless technology industry.

3.1.5 SDR Application

Through the last two decades of open source developing, the SDR has about several hundreds of applications such as Cognitive Radio, RF-ID and OpenBTS which is our project subject and we will talk about it in details later.

3.2 Universal Software Radio Peripheral (USRP)

The Universal Software Radio Peripheral (USRP) products are computer-hosted devices that support the use of SDR designed and sold by Ettus Research and its parent company, National Instruments. Developed by a team led by Matt Ettus, the USRP product family is intended to be a comparatively inexpensive hardware platform for software radio, and is commonly used by research labs, universities, and hobbyists.

The USRP is designed to allow general purpose computers to function as high bandwidth software radios. In essence, it serves as a digital baseband and IF section of a radio communication system.

In addition, it has a well-defined electrical and mechanical interface to RF front-ends (daughter boards) which can translate between that IF or baseband and the RF bands of interest.

The USRP does all of the waveform specific processing on the host CPU like

- Modulation and Demodulation

All of the high speed general purpose operations are done on the FPGA like

- Digital up Conversion (DUC).
- Digital down Conversion (DDC).
- Decimation.

3.2.1 UHD (USRP Hardware Driver)

USRP Hardware Driver is the device driver provided by Ettus Research for use with the USRP product family. It works on all major platforms Linux, Windows, and Mac.

The goal of UHD is to provide a host driver and API for current and future Ettus Research products. Users will be able to use the UHD driver standalone or with third-party applications such as:

- GNU Radio.
- LabVIEW.
- MATLAB.
- OpenBTS

3.2.2 Products

Table 3-1: Different types of USRPS

	Networked Series	Bus Series	Embedded Series
Models	N200/210	USRP1/B100/B200/B210	E100/E110/E310
Host interface	Gigabit Ethernet	USB 2.0/3.0	Embedded
MIMO capability	Exists	Doesn't exist	Doesn't exist
Host BW (MHz)	50	16	4-8
ADC	14 bit, 100 MSPS	12 bit, 64 MSPS	12 bit, 64MSPS
DAC	16 bit, 400 MSPS	14 bit, 128MSPS	14 bit, 128 MSPS

3.2.3 USRP Component

It consists of motherboard (brain of USRP) which provides the following subsystems:

- FPGA,
- ADCs, DACs,
- Host processor interface
- Power regulation.
- Clock generation and synchronization as well as conversion

These are the basic components that are required for baseband processing of signals.

Some USRP models connect to the computer using high speed gigabit Ethernet cables or using USB ports which the host-based software uses to control the USRP hardware and transmit/receive data.

The FPGA provides digital signal processing operations which provide translation from real signals in the analog domain to lower rate, complex, baseband signals in the digital domain. The code for FPGA is open-source and can be modified to allow high speed, low latency operations to occur in FPGA.

It also contains the daughterboard. The interchangeable daughter boards turn USRP motherboard into a complete RF transceiver system. Just add an antenna, and you are ready for two-way, high bandwidth communications in many popular frequency bands, it is used for analog operations such as up/down-conversion, filtering, and other signal conditioning. This modularity permits the USRP to serve applications that operate between DC and 6 GHz.

3.2.4 USRP N210

The USRP N210 as shown in [Figure 3-2](#) provides high-bandwidth, high-dynamic range processing capability. The USRP N210 is intended for demanding communications applications requiring this type of rapid development. The product architecture includes a Xilinx® Spartan® 3A-DSP 3400 FPGA, 100 MS/s dual ADC, 400 MS/s dual DAC and Gigabit Ethernet connectivity to stream data to and from host processors. A modular design allows the USRP N210 to operate from DC to 6 GHz, while an expansion port allows multiple USRP N210 series devices to be synchronized and used in a MIMO configuration.

USRP N210 specifications are described in [table 3-2](#) also the internal structure is described in [figure 3-3](#).



Figure 3-2: USRP N210

Table 3-2: USRP Specifications

Spec	Typ.	Unit	Spec	Typ.	Unit
POWER			RF PERFORMANCE (w/ WBX)		
DC Input	6	V	SSB/LO Suppression	35/50	dBc
Current Consumption	1.3	A	Phase Noise (1.8 GHz)		
w/ WBX Daughterboard	2.3	A	10 kHz	-80	dBc/Hz
CONVERSION PERFORMANCE AND CLOCKS			100 kHz	-100	dBc/Hz
ADC Sample Rate	100	MS/s	1 MHz	-137	dBc/Hz
ADC Resolution	14	bits	Power Output	15	dBm
ADC Wideband SFDR	88	dBc	IIP3	0	dBm
DAC Sample Rate	400	MS/s	Receive Noise Figure	5	dB
DAC Resolution	16	bits	PHYSICAL		
DAC Wideband SFDR	80	dBc	Operating Temperature	0 to 55°	C
Host Sample Rate (8b/16b)	50/25	MS/s	Dimensions (l x w x h)	22x16x5	cm
Frequency Accuracy	2.5	ppm	Weight	1.2	kg
w/ GPSDO Reference	0.01	ppm			

• All specifications are subject to change without notice.

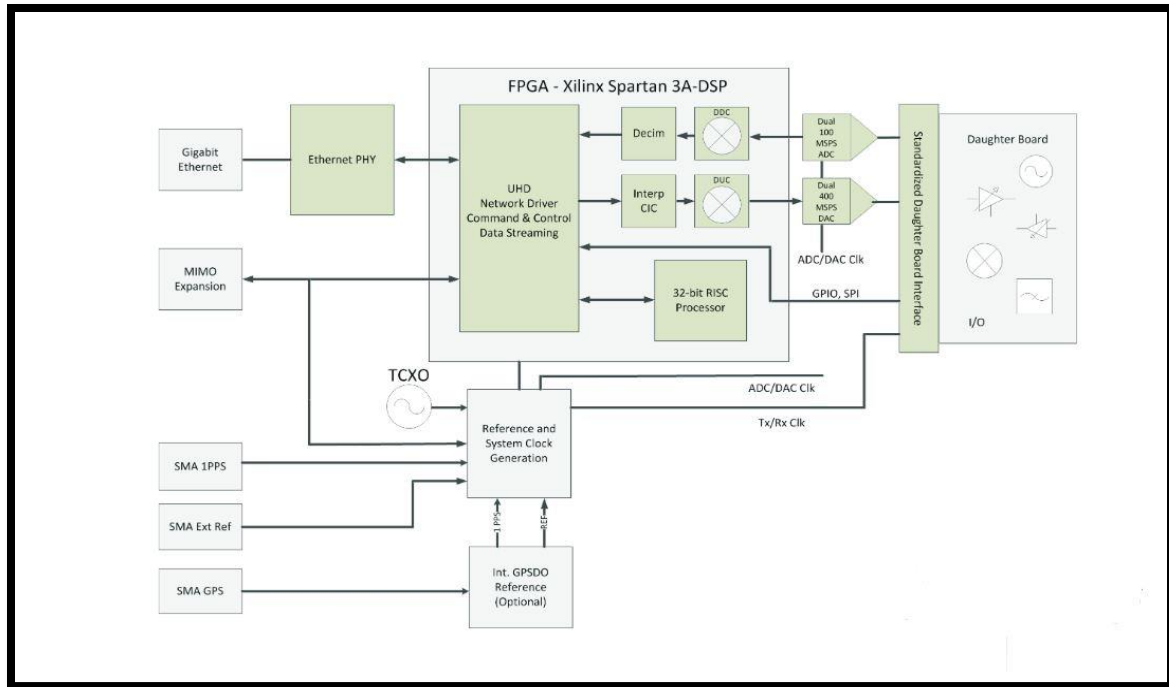


Figure 3-3: USRP N210 internal configuration

In our project we will use USRP N210 but Why N210 and not E100/E110? We choose N210 for 2 reasons:

- 1) practical issue happened before with previous graduation project their thesis book stated the following:

When we try to make a call between two mobile stations, sometimes it just rings and then fails to make a call and sometimes they couldn't even find the signal of the network.

- 2) Based on engineering concepts and numbers. It stated the following:

Table 3-1 shows the key characteristics of all USRP models available from Ettus Research. The table is useful for determining the interface type, bandwidth capabilities, and synchronization mechanisms specified for each USRP model. You can use this information, and the requirements for the application in question, to select a USRP radio.

We need to answer some questions to determine the best USRP to use:

- **Do I want to perform processing on a host PC, or operate the USRP device in a standalone fashion?**

This is an obvious differentiator of the USRP Embedded Series. If you need the USRP to operate a USRP radio without a host PC, the USRP E100/E110 is the most appropriate. The USRP E100/E110 is ideal for applications that might require mobile transceivers or distributed RF sensors. Unless the user has a clear requirement for embedded operation, Ettus Research recommends the USRP N200, N210, B100, or USRP1. Developing with a host-based platform typically involves less risk and will require less effort to optimize various pieces of the software radio.

- **Do I Need Synchronization and/or MIMO Capability?**

If you need MIMO capability for your application, Ettus Research recommends the USRP N200 or USRP N210. These units can be synchronized by providing a common time and frequency reference. Two USRP N200/N210s can be synchronized for MIMO operation with an Ettus Research MIMO cable. Alternatively, external 10 MHz reference and 1 PPS signals can be distributed to multiple USRP radios. With proper consideration for interface issues, it is possible to create MIMO system of arbitrary size with the USRP N200/N210.

- **What Are My Bandwidth Requirements?**

Many Bandwidth requirements can also be used to narrow down the USRP selection. As seen in the Table 3.1, the USRP N200/N210 is capable of streaming up to 50 MS/s in each direction in 8-bit mode, and 25 MS/s in 16-bit mode. If there is interest in transmit and/or receiving large bandwidth signals such as 802.11, the USRP N200/N210 would be more appropriate.

The USRP E100/110 FPGA interface provides a maximum throughput of 40 MB/s.

This bandwidth can be used distributed across transmit and receive sample transfer. At 4 bytes/sample, this provides for a total of 10 MS/s. Note this does not guarantee that the embedded processor will be able to process that many samples.

- **What interface do I prefer to work with?**

Assuming you have narrowed the viable devices down based on bandwidth, MIMO and channel count requirements, it is possible to select a USRP device based on the interface. The USRP N200/N210 requires a Gigabit Ethernet port and a PC typically only provides one such port. If internet access is required, the user will also need to plan for an additional network adaptor.

The Gigabit Ethernet interface of the USRP N200/N210 can operate over significantly longer ranges. This makes it possible to operate the USRP radio at more remote locations further from the host computer. The GigE interface can be accessed via a Gigabit Ethernet switch, allowing access to multiple devices. However, Ettus Research recommends a homogeneous network without other devices, such as network routers attached.

So we have decided to use N210 instead of E100/E110 because:

- 1) 1-total host Bandwidth in N210 is higher than that in E100 so it supports more users.
- 2) 2- MIMO capability exists in N210 not E100.
- 3) 3- N210 processing is higher.N210 supports UMTS.

3.2.5 Daughterboards

A daughterboard is a circuit attached to the motherboard, acting as an extension for performance specific functions. Different frequencies require different Antennas and sometimes different signal processing like amplification or filtering to transmit and receive correctly. There are several types of daughterboards: Transmitters, Receivers and transceivers. Transmitter modulate an output signal to a higher frequency (Carrier Frequency). Receiver acquires a RF signal and convert it to baseband (handles demodulation). Transceiver combines the functionality of a transmitter and Receiver. **Table 3-3** shows the list of available daughterboards.

Ettus Research recommends the WBX or SBX daughterboards, which provide wide frequency coverage illustrated in **figure 3-4**. So we choose WBX according to the band we need in our project.

Table 3-3: List of available daughterboards

Model	Type	Frequency range	BW (MHZ)	Power output (mw)	Noise figure
TVRX2	Rx	50-860 MHZ	10	N/A	4-10
RFX900	TX/RX Full-Duplex	750-1050 MHZ	30	200	5-10
RFX1800	TX/RX Full-Duplex	1.5-2.1 GHZ	30	100	5-10
RFX2400	TX/RX Full-Duplex	2.3-2.9 GHZ	30	50	5-10
WBX	TX/RX Full-Duplex	50 MHZ -2.2 GHZ	40	100	5-10
SBX	TX/RX Full-Duplex	400 MHZ-4.4 GHZ	40	100	5-10
XCVR2450	TX/RX Full-Duplex	2.4 GHZ- 2.5 GHZ	33	100	5-10
DBSRX2	RX	800 MHZ -2.35 GHZ	1-60	N/A	4-8
LFTX	2xTX	DC-30 MHZ	60	1	N/A
LFRX	2xRX	DC – 30 MHZ	60	N/A	N/A

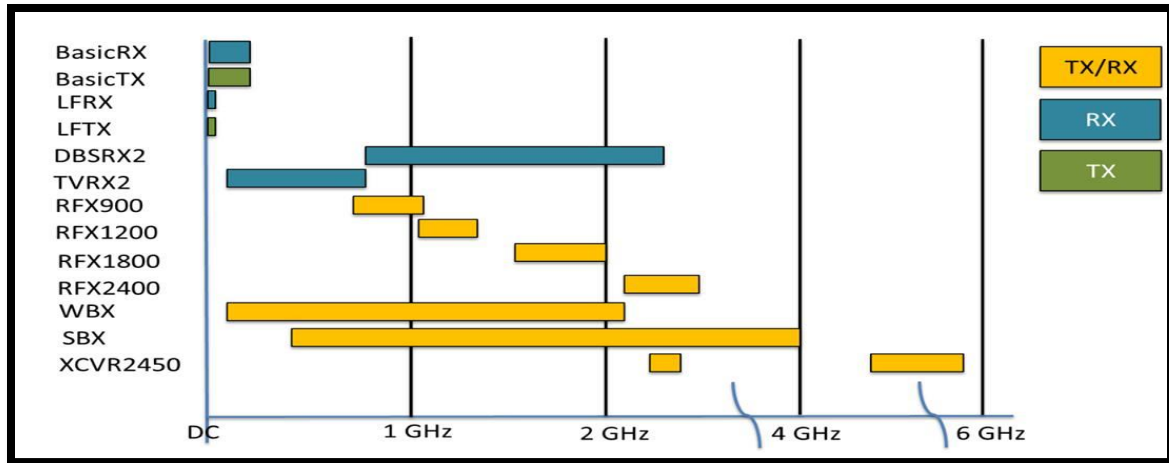


Figure 3-4: RF Daughterboard Frequency Coverage

3.2.6 Antennas






An Antenna is an electrical device that converts electrical power to radio signals and vice versa. They are used to send and receive data from several USRPs. They are used in many communication applications such as radio broadcasting, radars, cell phones, etc.

Table 3-4 shows the list of Antennas offered by Ettus Research and their specifications. We choose Vert900 in GSM emulation.

3.2.7 USRP Applications

The USRP product family is used all over the world in a wide variety of applications, while the USRP is often used for rapid prototyping and research applications, it has been deployed in many real world commercial and defense systems. There are many applications for the USRP in commercial systems as an example and referring to our project mobile access for aviation industry. System development and prototyping is done on software radio. The flexibility of the USRP enables a low cost system. The USRP can be configured to be used as a testing equipment, GSM base station, FM radio transmitter and receiver, passive radar, synthetic aperture radio, DAB transmitter and mobile WIMAX receiver. Besides it can be used to listen to AM/FM/ Aircraft and military radio bands and reading RFID tags and receiving slow scan television broadcasts from the international space station.

Table 3-4: List of available Antennas

Model	Operating Frequency range	shape
LP0410	400 MHz to 1 GHz	
LP0965	850 MHz to 6.5 GHz	
Vert400	144 MHz , 400 MHz And 1200 MHz	
Vert900	824 to 960 MHz And 1710 to 1990 MHz	
Vert2450	2.4 to 2.48 GHz And 4.9 to 5.9 GHz	

3.3 OpenBTS (Open Base Transceiver Station)

The construction of an actual base station is too time devouring and most importantly extremely expensive. This is where the idea of implementing a cost effective GSM base station came from. GSM services can now be available in rural areas, developing countries, and hard to reach locations such as oil rigs. Implementation of a GSM base station requires two main components, a Software Defined Radio (USRP N210) and OpenBTS.

OpenBTS is an open-source UNIX application that uses the Universal Software Radio Peripheral (USRP) to present a GSM air interface ("Um") to standard GSM handset and uses the Asterisk software PBX to connect calls. The combination of the ubiquitous GSM air interface with VoIP backhaul could form the basis of a new type of cellular network that could be deployed and operated at substantially lower cost than existing technologies in Greenfields in the developing world.

OpenBTS is a software-based GSM access point, allowing standard GSM-compatible mobile phones to make telephone calls without using existing telecommunication providers' networks. OpenBTS is notable for being the first free software implementation of the industry-standard GSM protocol stack. In other word OpenBTS = GSM + VOIP.

3.3.1 OpenBTS and traditional GSM

In this section we will know how OpenBTS replaced the GSM Network Component (as shown in [Figure 3-5](#)) which we have mentioned previously:

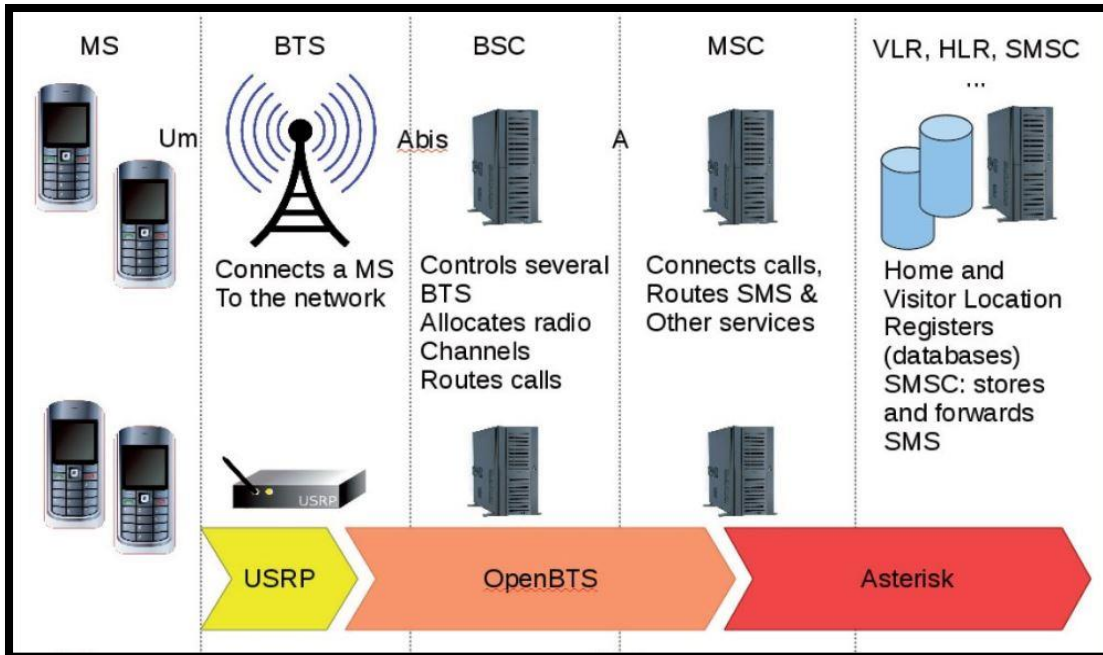


Figure 3-5: OpenBTS VS Traditional GSM

- 1) A USRP can be readily adapted as a GSM transceiver (BTS) (i.e.: it transmits and receives the GSM signal to and from the mobile phone).
- 2) OpenBTS software code which generates with UHD an air interface that to a cell phone, looks just like any other GSM cellular network. On the network side, it's an Asterisk server (VoIP), used to connect calls. OpenBTS software code plays the role of MSC/VLR in processing all the calls incoming to, or originating from subscribers visiting the given coverage area provided by the USRP's Antenna.

Using OpenBTS source code creates a beacon signal such that openBTS network is created and a phone can register to this network by manually setting the phone to search for alternative surrounding networks but, cannot make a phone call with another registered phone except when asterisk is installed and configured in this system. It allows attached phones to make calls to one another, and to connect to other phone services including PSTN and VOIP services. Asterisk plays the role of HLR in the traditional GSM network which

is the main database of permanent subscriber information for a mobile network (i.e.: it stores an IMSI for each subscriber, authentication key, subscriber status and the current location).

3.3.2 OpenBTS Advantage

The main advantage of the OpenBTS is the minimum cost as we can install the network at about 1/10 of the cost of current technologies, and still be compatible with most of the handsets that are already in the market. By replacing the GSM core network with commodity Hardware and open source Software. Also, OpenBTS allow bug fixing over the air or other remote reprogramming thus reducing both time and cost associated with operation and maintenance.

OpenBTS solves one of the toughest challenges for the Mobile Communication systems, which is the compatibility, as now it's about upgrading the software which is not comparable with Hardware replacement cost.

3.3.3 OpenBTS requirements

Hardware Requirements

- 1) A computer: Any normal PC can do the job
- 2) USRP N210: Used as Transceiver
- 3) Daughterboard: in this thesis a single WBX is used. It was chosen according to the GSM band you want to use.
- 4) Antenna: select an Antenna suited with the daughterboard .we use vert900 Antenna
- 5) Two unlocked Mobile phones equipped with SIM cards.

Software Requirements

- 1) A Linux Operating system (Ubuntu 14.04)
- 2) GNU Radio
- 3) SIP PBX such as Asterisk
- 4) OpenBTS software

3.4 OpenBTS application suite

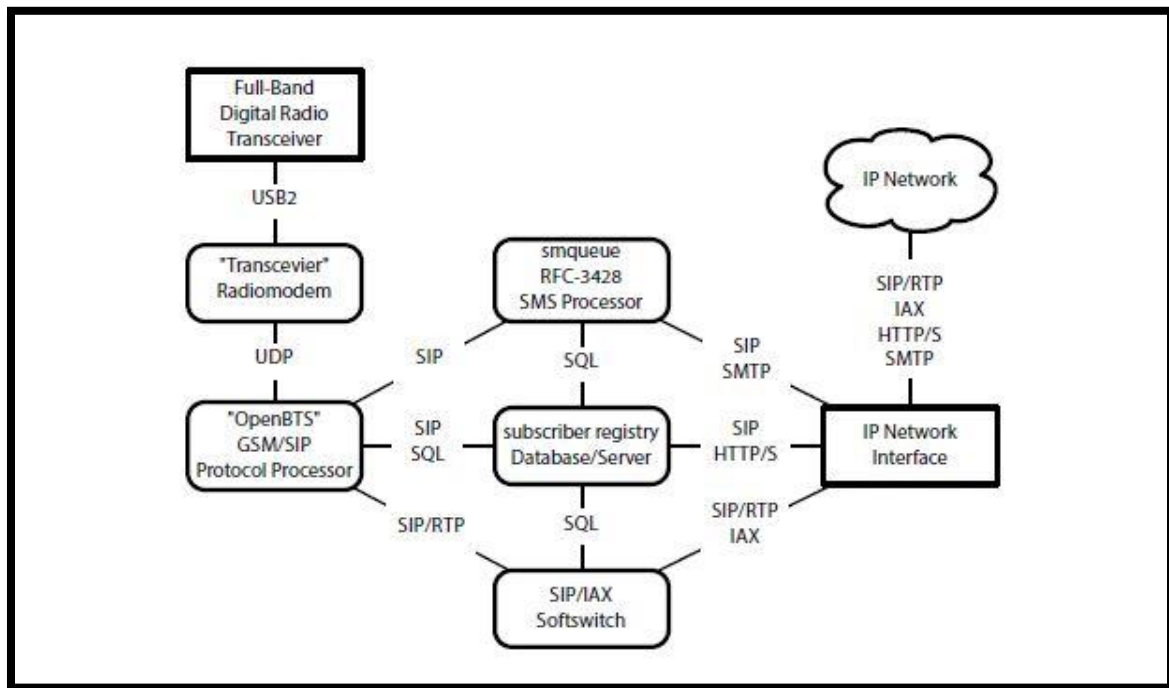


Figure 3-6: OpenBTS application suite

A complete OpenBTS installation comprises several distinct applications:

- OpenBTS - The actual OpenBTS application, containing most of the GSM stack above the radiomodem.
- Transceiver - The software radiomodem and hardware control interface.
- SMQueue - The RFC-3428 store-and-forward server for text messaging.
- Asterisk -The VoIP PBX or "softswitch".
- SIPAuthServe - An application managing the database of subscriber information.
- Other Services - Optional services supported through external servers, interfaced to OpenBTS through various protocols.

3.4.1 OpenBTS

OpenBTS is responsible for implementing the GSM air interface in software and communicating directly with GSM handsets over it. This communication is converted into SIP and RTP on the IP network side and interacts with the components above to form the core network. It implements the GSM stack above the radio modem.

3.4.2 Transceiver

The transceiver application performs the radiomodem functions of GSM and manages the USB interface to the radio hardware.

The Transceiver is responsible for transmitting and receiving samples to and from the USRP, also it passes these samples in the form of raw bits to the GSM stack in case of reception or receives them from the GSM stack in case of transmission. It interfaces with the GSM stack through UDP socket, and with the USRP (Full band Digital Radio Transceiver) through USB 2.0 or Ethernet. It performs the basic operations such as modulation, interleaving, correlation, etc.

3.4.3 SMQueue

SMQueue is an RFC-3428 store-and-forward server that is used for text messaging in the OpenBTS system. SMQueue is required to send a text message from one MS to another, or to provide reliable delivery of text messages to an MS from any source.

3.4.4 SIP router/PBX

OpenBTS uses a SIP router or PBX to perform the call control functions that would normally be performed by the mobile switching center in a conventional GSM network, although in most network configurations this switching function is distributed over multiple switches. These switches also provide transcoding services. As of OpenBTS Release 4.0, the standard SIP router is Asterisk 11.

3.4.5 SIPAuthServe

It stands for SIP Authentication Server .it is an application that implements Subscriber Registry which is the database of subscriber information that replaces both the Asterisk SIP registry and the GSM Home Location Register (HLR) found in a conventional GSM network. It is responsible for SIP registration and authentication server, used to process

location updating requests from OpenBTS and perform corresponding updates in the subscriber registry database.

3.5 OpenBTS application Protocols

Now, we need to understand different protocols and interfaces between different applications.

SIP (Session Initiation Protocol)

It is a protocol used for controlling multimedia communication sessions such as voice and video calls over Internet Protocol (IP) network.

UDP (User Datagram Protocol)

UDP is part of the Internet Protocol suite used by programs running on different computers on a network. UDP is used to send short messages called datagrams but overall, it is an unreliable, connectionless protocol.

RTP (Real-Time Transport Protocol)

It 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. RTP is one of the technical foundations of Voice over IP and is often used in conjunction with a signaling protocol such as the Session Initiation Protocol (SIP) which establishes connections across the network. RTP is designed for end-to-end, real-time, transfer of streaming media.

SMTP (Simple Mail Transfer Protocol)

SMTP is an Internet standard for electronic mail (email) transmission. SMTP is a connection-oriented, text-based protocol in which a mail sender communicates with a mail receiver by issuing command strings and supplying necessary data over a reliable ordered data stream channel, typically a Transmission Control Protocol (TCP) connection.

HTTP/S (Secure HyperText Transfer Protocol)

It is the secure version of HTTP, the protocol over which data is sent between your browser and the website that you are connected to. The 'S' at the end of HTTPS stands for 'Secure'. It means all communications between your browser and the website are encrypted. HTTPS is often used to protect highly confidential online transactions like online banking and online shopping order forms.

IAX (Inter-Asterisk eXchange)

It is a communications protocol native to the Asterisk private branch exchange (PBX) software, and is supported by a few other softswitches, PBX systems, and softphones. It is used for transporting VoIP telephony sessions between servers and to terminal devices.

SQL Server

As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications which may run either on the same computer or on another computer across a network (including the Internet).

Chapter 4

SkyComm solution for aviation industry

4.1 Motivation

In this chapter we will talk about the mechanism of calling between two mobile phones one of them in the plane and the other is not and we will identify the different phases and problems that face us to make the two mobile phones are attached to each other and how calling process is done successfully.

4.2 Safety design

The first question that should be asked is: Does any component affect the plane safety, plane navigation or communication with towers during flight? Or in other words, do we make interference to plane's electronic components during calls?

We have to show and prove that everything will work well and the answer of the two previous questions must be no. The navigation devices frequencies (according to Jneuhaus organization):

- Aircraft (Air carrier and Private), Aeronautical enroute “136.975 MHz”.
- Aircraft (Air carrier and Private), Airport control tower, Automatic weather observation “136.400-136.450 MHz”.
- Aircraft (Air carrier and Private), Flight test “123.575 MHz”.
- Aircraft (Air carrier and Private), Airport control tower, Aeronautical search and rescue “123.100 MHz”.
- Radio navigation land test “108.000 MHz”.
- Localizer “108.100-108.150 MHz”.

In the US the civil aircraft communications band (118-137 MHz) generally uses 25 kHz spaced channels. As of 2010 aeronautical enroute and flight test stations may use 8.33 kHz spaced channels in the 121.4-123.6, 128.825-132.0 and 136.5-136.875 MHz ranges.

Therefore no interference or damage may occur as the frequency is very far from that's used in plane as the minimum frequency used in mobile communication is 876 MHZ.

Now we are ready to talk about the phases of the solution in details after making sure that we are in the safe side of plane safety.

4.3 Solution overview

In order to solve the problem of applying mobile network on the plane we think critically to get the block diagram of our system. Firstly -because of the flight level- there are no mobile signal coverage in the plane, so we need an on air transceiver to apply this mobile signal coverage.

The next step is processing those signals and controlling the network to apply switching between calls, this step may be called base processing unit (BPU) which plays the rule of the core network and also contains the data base of the users of the network for authentication issues. Laptop with good processor and memory can model this

As we reach a laptop in BPU, we can easily use the IP network as a gateway of the network, and then we can connect to the satellite using the satellite unit which is implemented on the plane to reach the satellite link through the SATCOMM terminal.

The following figure shows the design flow of the SkyComm system and we are going to explain the procedure of the call in this chapter depending on this flow.

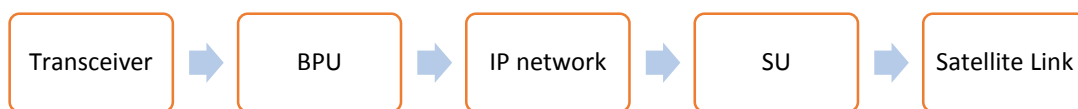


Figure 4-1: Design flow of SkyComm

4.4 Call flow during flight

4.4.1 Mobile Equipment

The mobile equipment has transceivers that can send and receive at 900, 1800, 1900, and 2100 MHz these frequencies support the 2G and 3G families. We will also choose the 1800 MHz as ETSI organization recommended as the minimum transmit power for a terminal in the 1800 MHz band is lower than in 900 MHz band (0 dBm instead of 5 dBm), and emissions at higher frequencies present higher path loss. The next step is the RF stage or antenna which supplies the signal to the mobile which will be selected to be leaky feeder for better coverage.

4.4.2 The leaky feeder

The leaky feeder is a coaxial cable that has small sections of its copper shielding stripped away to allow radio frequency (RF) signals to escape. Leaky feeders, which act as extended antennas, are also called radiating cables.

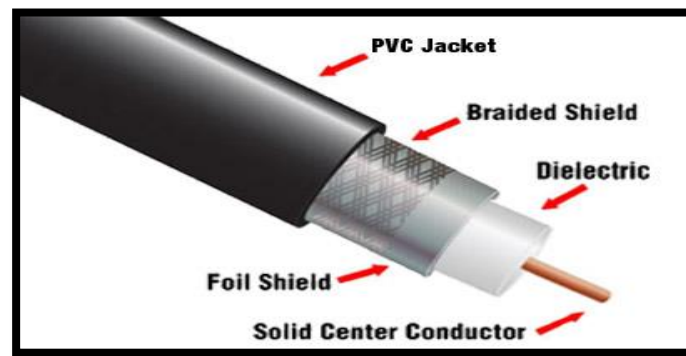


Figure 4-2: Leaky Feeder construction

Principle of operation

A leaky feeder communication system consists of a coaxial cable run along tunnels which emits and receives radio waves, functioning as an extended antenna. The cable is "leaky" in that it has gaps or slots in its outer conductor to allow the radio signal to leak into or out of the cable along its entire length. Because of this leakage of signal, line amplifiers are required to be inserted at regular intervals, typically every 350 to 500 meters, to boost the signal back up to acceptable levels. The signal is usually picked up by portable transceivers carried by personnel. Transmissions from the transceivers are picked up by the feeder and carried to other parts of the tunnel, allowing two-way radio communication throughout the tunnel system.

The system has a limited range and because of the frequency it uses (typically VHF or UHF), transmissions cannot pass through solid rock, which limits the system to a line-of-sight application. It does, however, allow two-way mobile communication as duplexers used to allow the cable to carry other frequencies at the same time, so the same cable can provide extended mobile phone coverage with 900 and 1800 MHz signals being carried simultaneously.

Applications

Leaky feeders are used in places where the actual structure makes RF communication difficult. Leaky feeders are often used in structures with metal frameworks such as skyscrapers, tunnels, ships and planes to extend mobile coverage. They are also useful in situations where low power levels are required to prevent interference with wireless microphones or other communication technologies that share the same frequency spectrum.

a) In-flight wireless networks

Leaky feeder antenna system can also be used to allow reception of on-board GSM and Wi-Fi signals on passenger aircraft. The weight and space requirements of leaky feeder systems are usually lower than comparable antenna systems, thus saving space and fuel. The even field strengths produced by runs of leaky feeders spanning the entire fuselage improve coverage while requiring less transmitting power.

b) Mining

Leaky feeder has been used in the mining industry as a method of wireless communication between miners. The system is used as a primary communication system which has a transceiver small enough to be comfortably worn on a miner throughout an entire shift.

c) Underground railways

Leaky feeder system is also used for underground mobile communication in mass transit railways.

d) Industrial buildings

Leaky feeder is also being used in warehouses and other industrial buildings where it is difficult to get Wi-Fi coverage using normal access points. Real life installations with 50–75 meters of leaky wire connected to the antenna input of Access Points exist, and are working fine.

Basic performance parameters

The performance of a leaky feeder system may be characterized by two parameters:

- a) Longitudinal attenuation
- b) Coupling loss

The longitudinal attenuation is governed primarily by the factors which apply to normal transmission lines, such as construction, conductor size and dielectric. Additionally there is a small loss component attributable to the leakage (or mode converters)

The coupling loss is, in general terms, the power loss between the feeder and a mobile antenna in its vicinity. For the commonly used coaxial types of leaky cable it is dependent on the degree of shielding in the feeder construction, the configuration of the shield or conductors and the permittivity of the dielectric. For a given cable construction it should also be noted that the coupling loss is also dependent upon:

- 1- The environment in which the cable is mounted
- 2- The cable mounting position

- 3- The characteristics, position and orientation of the mobile antenna
- 4- The operating frequency.

Types of leaky feeders

- a) Bifilar lines
- b) Continuously leaky coaxial cables
- c) Coaxial cables with periodic apertures
- d) Cables with mode converters.

Types a) and b) are intrinsically non-radiating in the sense that a cable of infinite length extending in free space can only carry waves guided by the structure. However, any discontinuity along the cable causes mode conversion and radiation.

In type c), the periodic apertures are radiating discontinuities and act like elements of an antenna array. Maximum radiation is obtained in oblique directions determined basically by the ratio of the spatial interval to the wavelength.

In type d), the mode converters or radiating elements are separate discontinuities acting in isolation

Example

GORE™ Leaky Feeder Antennas: Antennas provide consistent connectivity across a broad frequency range — from 400 megahertz up to 6 gigahertz — making the antennas compatible with numerous communication standards.

Typical applications:

- a) Wide-body aircraft.
- b) Single-aisle aircraft.
- c) Picocells for phone coverage.
- d) Wi-Fi 802.11 a/b/g/n/ac and WiMAX.
- e) Connectivity to Bluetooth, DECT, DECT2, Globalstar, GSM, IRIDIUM Sat, MMS, PDC, and TETRA protocols.

4.4.3 The ANC (Active Noise Cancellation)

The Active Noise Cancellation it is already located at mobile phones which generate a broadband noise floor which is being emitted through existing leaky line antenna masking reception, they measure and ensure that handsets can only connect to on board GSM network and will then operate with the lowest possible transmission power level GSM-1800 power control level its nominal output power of 0 dBm. This will result in significantly lower radiation levels than those experienced on average when using a mobile phone with terrestrial networks on ground. It is also used in safety from the interference of the airplane devices.

ANC is connected with the leaky line antenna, the Pico Cell or USRP and the server (if exists).

4.4.4 Core Network

The core network is the block which control and manage the network, switch between calls and containing the database of users to authenticate before establish calls.

For SkyComm solution the core network consists of USRP and laptop, USRP as an OpenBTS is a software-based GSM access point, allowing standard GSM-compatible mobile phones to be used as SIP endpoints in Voice over IP (VOIP) networks used in many telecommunication usages especially in air craft. It is very small in size about a hand palm size and the system uses Ethernet cabling for satellite linkage. It is available for most cellular technologies including GSM, CDMA, UMTS and LTE. It is responsible on the RF domain, its main rule is to apply GSM signal through leaky feeder; considering the frequency band, the modulation techniques, number of channels, multiplexing technique, bandwidth, time slot duration, etc.

Laptop is responsible for all processing in the network; it assigns channel and time slot to the users, takes decisions of assigned frequencies, power control, allocation and release of time channels, handover commands, synchronization, time advance, switching between users and call set-up procedure. From the other side it plays the rule of gateway of the network to the outside world through IP network by connecting to the satellite unit (SU). The detailed rule of USRP and OpenBTS is described before in chapter 3.

4.4.5 Satellite unit (SU)

Satellite unit consists of 4 components as following:

1) Satellite data unit:

A satellite data unit (SDU) is an avionics device installed in an aircraft that allows air/ground communication via a satellite network. It is an integral part of an aircraft's SATCOM (satellite communication) system. The device connects with a satellite via ordinary radio frequency (RF) communication and the satellite then connects to a ground station or vice versa. All satellite communication whether audio or data is processed by the SDU.

The SDU communicates with an on board MDDU (multi-purpose disk-drive unit) which maintains an updatable table of ground stations in the aircraft current area and the order of preference for selection of which ground station to use, which guides the choice of satellite. Along with analyzing data continuously sent from all ground stations (such as station status and the error rate of signals from each station) the SDU receives information about the aircraft's position and orientation from another on board system (ADIRU, air data inertial reference unit) which it passes to the BSU (beam-steering unit) to direct the signal beam from the aircraft to the chosen satellite.

The SDU complies with the latest ARINC 781 industry standard for Inmarsat SATCOM capability for classic Aeronautical, Swift64, and Swift-broadband operations. Even though the SDU is one of the lightest, smallest and most affordable SATCOM on the market, it is designed to maximize efficiency and scalability. The Satellite Data Unit (SDU) has a built-in amplifier or, depending upon installation constraints, it can be paired with an external Flange Mounted Power Amplifier (FMPA). The SDU is designed to work with the latest Inmarsat-approved High Gain Antenna systems (HGA) including a Duplexer Low Noise Amplifier (DLNA), and a Configuration Module (CM).

The fact that some companies provide the HGA-2100, DLNA-2100, and HCM-2100 ensures ease of installation of your complete SATCOM system, not to mention a single source for service when needed. Each Software Defined Radio (SDR) channel card performs failure reversion to the required cockpit safety (Classic Aero) function. The

system is designed to maintain communications without pilot intervention during all flight phases. Handover between satellites and spot beams is accomplished automatically, based on preferences contained in the Secure ORT as defined in ARINC 781.

2) Satellite modem:

A satellite modem or SatModem is a modem used to establish data transfers using a communications satellite as a relay.

There is a wide range of satellite modems from cheap devices for home Internet access to expensive multi-functional equipment for enterprise use.

A "modem" stands for "modulator-demodulator". A satellite modem's main function is to transform an input bit stream to a radio signal and vice versa. There are some devices that include only a demodulator (and no modulator, thus only allowing data to be downloaded by satellite) that are also referred to as "satellite modems." These devices are used in satellite Internet access case uploaded data is transferred through a conventional PSTN modem or an ADSL modem).

3) Satellite dish

A satellite dish is a dish-shaped type of parabolic antenna designed to receive electromagnetic signals from satellites, which transmit data transmissions or broadcasts, such as satellite television.

Principle of operation:

The parabolic shape of a dish reflects the signal to the dish's focal point. Mounted on brackets at the dish's focal point is a device called a feed horn. This feed horn is essentially the front-end of a waveguide that gathers the signals at or near the focal point and 'conducts' them to a low noise block down converter or LNB. The LNB converts the signals from electromagnetic or radio waves to electrical signals and shifts the signals from the down linked C-band and/or Ku-band to the L-band range. Direct broadcast satellite dishes use an LNBF, which integrates the feed horn with the LNB. (A new form of Omni directional satellite antenna, which does not use a directed parabolic dish and can be used on a mobile platform such as a vehicle was announced by the University of Waterloo in 2004.

The theoretical gain (directive gain) of a dish increases as the frequency increases. The actual gain depends on many factors including surface finish, accuracy of shape, feed horn matching. A typical value for a consumer type 60 cm satellite dish at 11.75 GHz is 37.50 dB.

With lower frequencies, C-band for example, dish designers have a wider choice of materials. The large size of dish required for lower frequencies led to the dishes being constructed from metal mesh on a metal framework. At higher frequencies, mesh type designs are rarer though some designs have used a solid dish with perforations.

A common misconception is that the LNBF (low-noise block/feed horn), the device at the front of the dish, receives the signal directly from the atmosphere. For instance, on BBC News down link shows a "red signal" being received by the LNBF directly instead of being beamed to the dish, which because of its parabolic shape will collect the signal into a smaller area and deliver it to the LNBF.

Modern dishes intended for home television use are generally 43 cm (18 in) to 80 cm (31 in) in diameter, and are fixed in one position, for Ku-band reception from one orbital position. Prior to the existence of direct broadcast satellite services, home users would generally have a motorized C-band dish of up to 3 m in diameter for reception of channels from different satellites. Overly small dishes can still cause problems, however, including rain fade and interference from adjacent satellites.

4) Very small aperture antenna

A very small aperture terminal (VSAT) is a two-way satellite ground station with a dish antenna that is smaller than 3 meters. The majority of VSAT antennas range from 75 cm to 1.2 m. Data rates range from 4 kbit/s up to 16 Mbit/s. VSATs access satellites in geosynchronous orbit to relay data from small remote earth stations (terminals) to other terminals (in mesh topology) or master earth station "hubs" (in star topology).

VSATs are used to transmit narrow band data (e.g., point of sale transactions using credit cards, polling or RFID data, or SCADA), or broadband data (for the provision of satellite Internet access to remote locations, VOIP or video). VSATs are also used for transportable,

on-the-move (utilizing phased array antennas) or mobile maritime communications. After passing through SU, the next step is reaching the satellite by a specified link.

4.4.6 Satellite Link

The only outlet of aeronautical communication is the satellite link, navigations, communication to ground stations or entertainment services such as internet or Aircomm services must pass through satellite link by SATCOM terminal which is fixed on the top of aircraft and connected to satellite data unit (SDU) with RF cable.

Satellite link is not ideal, there are some obstacles will come up against the sent data like fading, path loss and latency. Those obstacles must be studied well to be considered in the communication system to transmitter, to study those obstacles we need to study link properties of satellite from the aircraft represented in the terminal to the satellite then back from the satellite to the earth station and vice versa.

The first question must be asked which satellite orbit which serves the aeronautical communication services? Let's present a bit of peels about satellite communication system first then compare between the different types of orbits which can be used to be able to answer on this question.

The main utilize of satellite is for long distance communication because of the greater coverage area, independency on the distance and the higher bandwidth of satellite link but on the other hand it suffers from the large propagation delay.

One of the basic factor in satellite communication is the elevation angle which is the angle formed by the line of sight (The center of the satellite transmission beam) and the horizontal plane for an object above the horizontal plane as shown in [figure 4-3](#). Elevation angle affects the satellite coverage area.

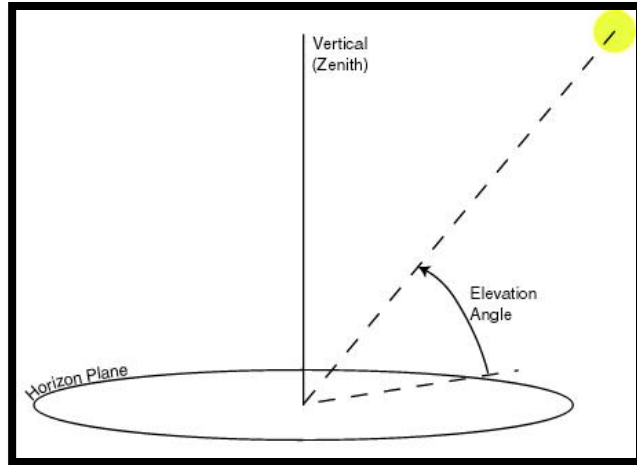


Figure 4-3: Elevation angel illustration

There are obstacles face the satellite communication, one of them is the attenuation which occur because atmosphere represented in rain and cloud and absorb some known frequencies, attenuation is also a function of elevation angle.

Now, we are ready to compare between the satellite orbits. There is no need to talk about the elliptical orbits; circular orbits are enough in case of aeronautical communication. There are three levels of circular orbits, Geostationary orbit (GEO), medium earth orbit (MEO) and low earth orbit (LEO).

GEO is in orbit around 36 Km above earth's surface and remain in the same position relative to the observer on surface of earth so in case of fixed object, the antenna will not track the satellite. But in case of aircraft the antenna must track the satellite but not rapidly because of the large coverage area and this reduces the Doppler's effect and handovers for the aircraft. On the other hand it has large propagation delay and needs high transmit power.

LEO is much closer to the earth, ranging from 500 to 1500 Km above earth's surface. It's not at fixed position relative to the surface. It's better in signal strength and time delay but has lower coverage area and the aircraft will suffer from Doppler shifts and handovers especially with the high speed of the airplanes.

MEO is between 8000 and 18000 Km above the earth surface and it is tradeoff between GEO and LEO solutions in issues like coverage areas and handovers.

Actually the main three orbits GEO, MEO and LEO can serve the aircraft, but each one has special advantages and special problems. But generally the lower orbits can reduce system capacity limitations and latency for real-time communication when the higher orbits reduce the system cost and network complexity such as reducing the handovers and offer higher coverage area. Note that elevation angle must have minimum value, under this value the transmission can't be occurs due to the attenuation.

The most practical solution in our days is the MEO orbit because it offers good coverage, not bad latency and moderate system complexity and cost. In the following we will first study the link properties between the aircraft and MEO satellite then between earth station and MEO satellite.

To get the performance of the link we need to calculate the carrier to noise ratio, which differs between uplink and downlink. Carrier to noise ratio is dependent of the receiver gain, wavelength, saturation flux density and noise temperature, but what is the noise temperature in satellite communication?

Noise temperature is a kind of noise which found by the resistance of the satellite antenna which can establish noise power, the temperature noise depends on the temperature and bandwidth. It is calculated by value called figure of Merit which characterize the ratio between gains to total noise temperature.

There are many other kinds of noise like Galactic noise which happens due to radiation from stars and planets and this kind of noise varies inversely with frequency, another kind is inter modulation noise which occurs where multiple carriers pass through any nonlinear device like travelling wave tube high power amplifier.

Now we can take a pick of link properties in satellite communication with aircraft and earth station.

Link properties between the aircraft and MEO satellite:

The satellite link frequency between the aircraft and the satellite is at Ku band and above because the lower frequencies are too limited for multimedia applications. The following

figure shows the satellite frequency bands. The aeronautical channel has been investigated in K band at 18.685 GHz.

During normal flight the aeronautical channel has been reported to have constant power with small fading when antenna is in line of sight with satellite because there are no obstacles like buildings. Fading up to 13 dB occurs due to shadowing or diffraction from the aircraft structure especially at low elevation areas.

Signal may suffer also from attenuation due to atmosphere due to cloud, rains, vapor and oxygen. Atmospheric attenuation depends on the flight altitude, the region, and the weather conditions.

The link properties for aircraft are not the same in different aircraft scenarios like parking, takeoff, landing and flight.

LETTER DESIGNATION FOR SATELLITE FREQUENCY BAND	FREQUENCY RANGE (GHZ)
L	1 - 2
S	2 - 4
C	4 - 8
X	8 - 12 (8 - 12.5 in North America)
Ku	12 - 18 (12.5 - 18 in North America)
K	18 - 27 (18 - 25.5 in North America)
Ka	27 - 40 (26.5 - 40 in North America)
O	40 - 50
V	50 - 75

Figure 4-4: Satellite Bands

Link properties between the Earth Station and MEO satellite:

The channel between the satellite and the earth station is different than the aeronautical channel because it is farther and may suffer from more obstacles for an example the path loss can completely change very quickly as a user moves from a clear state with a line of sight to a blocked state in a building or mountain but it doesn't suffer from handovers like the aeronautical one.

Satellite communication to earth has multipath effect and fading caused by buildings, hills and other obstacles on the ground.

Other factors of link properties are path loss and attenuation which are increasing with the frequency and increasing with the elevation angle decreasing. Propagation impairments caused by the natural medium must be included in the characterization of the link channel between satellite and earth station, some of this impairments as mentioned in Suzuki model are:

- 1) Absorption of the atmosphere is lower than 1 dB and decreases as the elevation angle increases.
- 2) Attenuation caused by rain is proportional to the strength of rain but always less than 0.5 dB.
- 3) Attenuation caused by cloud or fog is less than 0.03 dB.
- 4) Attenuation caused by snow is less than 0.01 dB.
- 5) Reflection by atmosphere is less than 0.2 dB and happens when elevation angle is higher than 5 degrees.
- 6) Polarization effects make the loss reach 9 dB at specific frequencies.

4.4.7 How to reach mobile phone from Earth station?

A ground station, earth station, or earth terminal is a terrestrial radio station designed for extra planetary telecommunication with spacecraft (constituting part of the ground segment of the spacecraft system), or reception of radio waves from astronomical radio sources.

Ground stations may be located either on the surface of the Earth, or in its atmosphere. Earth stations communicate with spacecraft by transmitting and receiving radio waves in the super high frequency or extremely high frequency bands (e.g., microwaves). When a ground station successfully transmits radio waves to a spacecraft (or vice versa), it establishes a telecommunications link. A principal telecommunications device of the ground station is the parabolic antenna.

Earth station receives signal from satellite with very high frequency and need to convert signal to different destinations like mobiles , TV & and many applications. IN our subject we will deal with how to reach mobile phone?

Earth station after receiving the call there are many cabinets which have many roles one of them it's role how to reach mobile phones by using radio optical frequency (Rof) to reach public switched telephone network (PSTN) then go to gateway by means of communication and the call is treated as a normal call which we have talked about in chapter 2 (GSM) As shown below satellite uplink and downlink block diagrams.

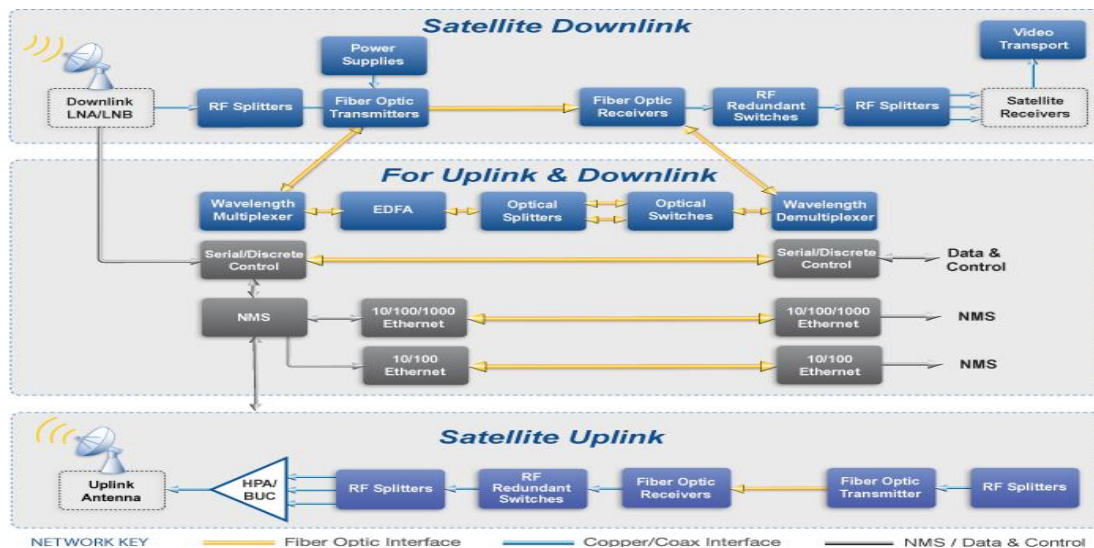


Figure 4-5: Uplink & downlink flow

Earth Station Components

An Earth terminal will always include an antenna. Terminals may also include:

- 1) Low-noise amplifiers (LNAs) or low-noise down-converters
- 2) High-power amplifiers (HPAs)
- 3) Signal processing equipment (e.g. down-converters, up-converters, IF amplifiers, modems and codecs)
- 4) Transmission and signaling equipment at the interface between the terminal and the terrestrial network

- 5) Supervisory and control equipment enclosures to protect the equipment from the environment

Note: Not everything is present in every terminal. For example, HPAs are not required for receive-only terminals.

Radio optical fibers

Radio over fiber (RoF) refers to a technology whereby light is modulated by a radio signal and transmitted over an optical fiber link to facilitate wireless access, such as 2G, 3G and Wi-Fi simultaneous from the same antenna. In other words, radio signals are carried over fiber-optic cable. Thus, a single antenna can receive any and all radio signals (2G, 3G, Wi-Fi, etc..) carried over a single-fiber cable to a central location where equipment then converts the signals; this is opposed to the traditional way where each protocol type (2G, 3G, Wi-Fi) requires separate equipment at the location of the antenna.

The Structure of Optical Fiber

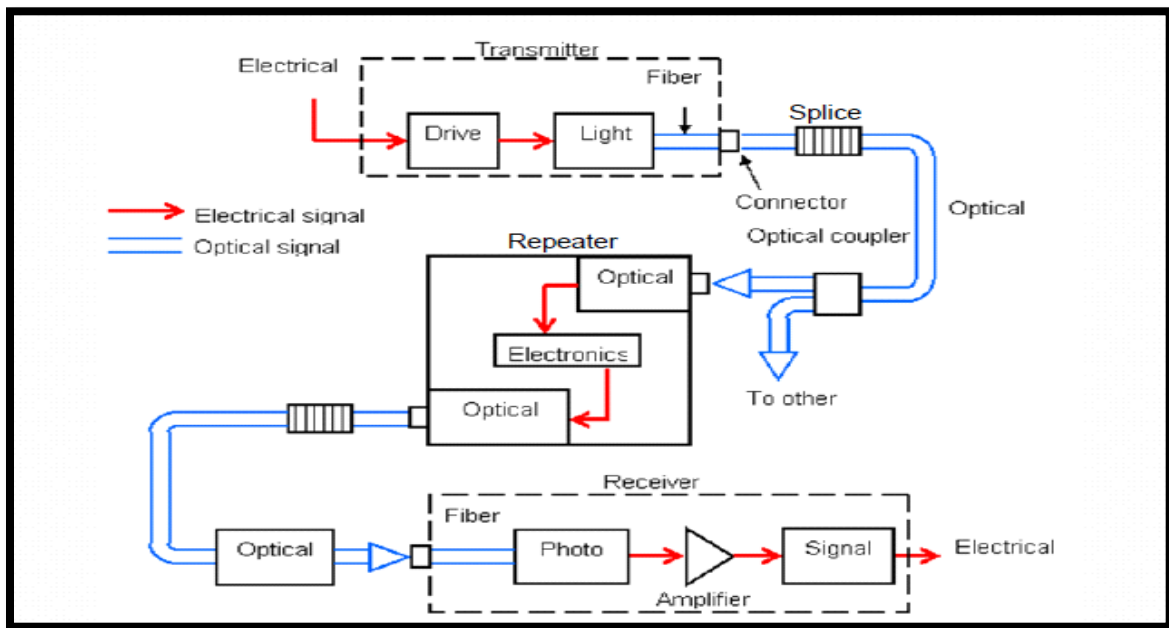


Figure 4-6: Optical Fiber Communication

Chapter 5

Installation

In this chapter, we will guide you through the selection of hardware, installation of a base operating system, and development environment setup, as well as actually compiling and installing the components that compose the OpenBTS software suite. Several shortcuts can be taken along the way—for example, if you would prefer to use the official binary packages instead of compiling your own—but the entire process is detailed for those wishing to build from scratch.

5.1 Hardware component

Although OpenBTS implements most of the complexity involved in building a mobile network in software, radio waves must still be transmitted and received somehow. This section details which hardware components you should procure to implement this capability in a development setting.

5.2 Linux Server

The first requirement is a standard commodity Linux server. Other architectures are beginning to be supported, but stick to an x86 processor running a 32-bit operating system for the best results for now. This computer can be a separate machine in your test environment or it can actually be a virtual machine on the laptop or desktop you use daily.

Minimum requirements for processing power and RAM are not clearly defined due to the many variables involved, such as the number of concurrent carrier signals, network load, network usage type, radio environment, etc. Each will affect the required resources.

A single carrier signal requires the OpenBTS software to generate downlink waveforms to transmit to the handset and demodulate the uplink waveforms received from the handset. OpenBTS supports the creation of multiple concurrent carrier signals on a single physical radio to linearly increase network capacity, but the processing demands are very high. For a stable lab setup with a single carrier signal (maximum of seven concurrent voice

channels), an Intel i5 or something comparable with 2 GB of RAM is recommended. It must also have at least a USB2 interface, but USB3 is quickly becoming a requirement on newer software defined radios. It may be a good idea to start with USB3 now to avoid a future upgrade. In our actual installation we are using N-series of USRP so we deal with Ethernet connectivity.

The need for increased throughput on the USB interface relates to the quantity and size of radio waveform samples being communicated through it. In a production environment, multiple simultaneous carrier signals can be utilized, which drastically increases the required sample bandwidth. Additionally, the algorithms used to demodulate signals can be configured to operate in a more robust manner (e.g., to rectify signal distortion due to the deployment environment). Thus, the processing power needed to generate and demodulate these signals could be an order of magnitude greater than in a lab setup.

Another thing to keep in mind is that as newer releases of OpenBTS are made available, new capabilities may require more processing power or memory. For example, OpenBTS in Global System for Mobile (GSM) mode can run smoothly on an Intel Atom processor but the new OpenBTS-UMTS (Universal Mobile Telephone System) requires at least an Intel i7. This side note is actually the main advantage of having a radio access network (RAN) defined in software. As new standards are released, they can be applied to a production RAN via a simple software update instead of swapping expensive hardware infrastructure.

5.3 Software Defined Radio

The software defined radio (SDR) is the key breakthrough that makes OpenBTS possible from a hardware perspective. SDRs have been used in military applications for about 20 years. Only recently have they become available to a wider audience due to the decreasing cost of the technology.

OpenBTS supports SDRs from several vendors: Ettus Research, Fairwaves, Nuand, and Range Networks. These products range in price from approximately \$500 to over \$2500.

If you choose a product that connects via Ethernet, make sure your Linux server has a dedicated Ethernet port for the radio (preferably gigabit Ethernet). Most SDRs are

completely generic hardware suitable for any radio project while some are specifically optimized for implementing mobile networks. Check with the vendor before purchasing.

5.4 Antennas

Many SDRs have enough transmit and receive sensitivity to operate without antennas in a small environment. Typically, a coverage area with a radius of 1 m can be achieved. This is a desirable setup for a lab environment, especially if multiple developers are using multiple radios. The coverage areas will not overlap and interfere with each other. Additionally, your network will not interfere with any carriers in the area.

In lab environments where a single OpenBTS instance is to be shared among multiple developers, the coverage area must be expanded. Adding a pair of small 5 dBi antennas can increase it dramatically, up to a 25 m radius in an unobstructed environment. These are usually rubber duck–style antennas with a subminiature version A (SMA) connector, similar in appearance to a typical home Wi-Fi router antenna. An example is shown in [Figure 5-1](#).



Figure 5-1: Rubber-duck antenna

You must also make sure that the handset you are using is unlocked. If the handset is “locked,” it means that the manufacturer has programmed the hardware’s baseband processor to only work with a specific carrier. This restriction can be removed, usually by

entering a sequence of numbers on the dial pad, but that is beyond the scope of this book. The easiest choice is to use an unlocked handset that will accept any carrier's subscriber identity module (SIM) card.

5.5 Operating System and Development Environment Setup

Now that the hardware has been gathered, you are ready to proceed with setting up a development environment. OpenBTS has traditionally been developed and tested on Ubuntu Long-Term Support (LTS) distributions. It has also been tested on Debian and CentOS distributions.

For this book, the most tested distribution and architecture will be used: Ubuntu 14 LTS 64-bit. Starting with OpenBTS 5.0. In addition to Ubuntu 12, the Ubuntu 13 systems have also been used successfully. Preliminary packaging for RPM-based systems (CentOS, Fedora, and Red Hat Enterprise Linux) is also available but will not be covered in this book.

5.6 Git Compatibility

Git is a version-control system that manages software source code changes. The OpenBTS project utilizes several new features in Git, such as sub module branch tracking. To make sure your client is compatible (e.g., newer than 1.8.2), it needs to be updated.

First, execute this command to add support for Personal Package Archives, an alternate way to distribute binary release packages:

```
$ sudo apt-get install software-properties-common python-software-properties
```

Then, execute the following command to add a repository for the latest Git builds to your system:

```
$ sudo add-apt-repository ppa:git-core/ppa
```

Now, you must simply refresh the list of packages and install Git again to update your system's client:

```
$ sudo apt-get update
```

```
$ sudo apt-get install git
```

To confirm that the new Git client is installed properly, run the following command:

```
$ git --version
```

Now that you have Git installed, you can proceed to downloading the development scripts.

5.7 Downloading the Code

The OpenBTS project consists of multiple software components hosted in separate development repositories on GitHub. Understanding the intricacies of Git should not be a barrier to using OpenBTS, so several development scripts have been written to make it easier to download the code, switch branches, and compile components. To download these development scripts into your new environment, run the following command:

```
$ git clone https://github.com/RangeNetworks/dev.git
```

The development scripts assume that you have Secure Shell (SSH) keys set up for GitHub. If you do not, please follow these instructions to set them up before proceeding:

Checking for existing SSH keys:

- 1- Open Git Bash.
- 2- Enter `ls -al ~/.ssh` to see if existing SSH keys are present:

```
$ ls -al ~/.ssh
```

Lists the files in your .ssh directory, if they exist
- 3- Check the directory listing to see if you already have a public SSH key.

By default, the filenames of the public keys are one of the following:

- id_dsa.pub
- id_ecdsa.pub
- id_ed25519.pub
- id_rsa.pub

If you don't have an existing public and private key pair, or don't wish to use any that are available to, generate a new SSH key.

Generating a new SSH key:

- 1- Open Git Bash.
- 2- Paste the text below, substituting in your GitHub email address.

```
$ ssh-keygen -t rsa -b 4096 -C "your_email@example.com"  
# Creates a new ssh key, using the provided email as a label  
Generating public/private rsa key pair
```

- 3- When you're prompted to "Enter a file in which to save the key," press Enter. This accepts the default file location.
Enter a file in which to save the key (/Users/you/.ssh/id_rsa): [Press enter]
- 4- At the prompt, type a secure passphrase.

Adding your SSH key to the ssh-agent:

- 1- Ensure ssh-agent is enabled:

```
# start the ssh-agent in the background  
$ eval "$(ssh-agent -s)"  
Agent pid 59566
```
- 2- Add your SSH key to the ssh-agent.

```
$ ssh-add ~/.ssh/id_rsa
```

Now, to download all of the components, simply run the clone.sh script:

```
$ cd dev
```

```
$ ./clone.sh
```

Each component's repository will be cloned from GitHub into your development environment. The clone.sh script also automatically initializes any sub modules needed.

Now that the OpenBTS project sources are in your development environment, you can select a specific branch or release to compile. The `switchto.sh` script is used to toggle between build version targets. For example, if you wanted to build the v4.0.0 release, run the following command:

```
$ ./switchto.sh v4.0.0
```

The current version target can be listed for each component by using the `state.sh` script. This script also lists any outstanding local changes for each component.

```
$ ./state.sh
```

This book focuses on the 5.0 series branch. To target the latest, greatest code in 5.0, run the following command:

```
$ ./switchto.sh 5.0
```

5.8 Building and installing the Code

Building the code

Your development environment is now prepared to build the newest bits in the 5.0 series branch. To compile binary packages, you will use the `build.sh` script. It automatically installs the compiler and autoconfiguration tools as well as any required dependencies. It also controls which radio transceiver application will be built. As there are several different drivers available for the various radio types, `build.sh` requires an argument so it knows which hardware is being targeted (valid radio types are SDR1, USRP1, B100, B110, B200, B210, N200, and N210).

This book targets the Ettus Research N210. Run the build command now:

```
$ ./build.sh N210
```

This process can take a while (30–60 minutes) the first time it is run, depending on the hardware it is being executed on. Going forward, you will only need to recompile updated components. The following command, for example, recompiles just the OpenBTS package for an Ettus Research N210 radio:

```
$. /build.sh N210 openbts
```

If UHD is not found, the following commands will be used:

```
$ sudo apt-get install libboost-all-dev libusb-1.0-0-dev python-mako doxygen python-docutils cmake  
build-essential libncurses5 libncurses5-dev  
  
$ git clone git://github.com/EttusResearch/uhd.git  
  
$ cd <uhd_directory>/host  
  
$ mkdir build  
  
$ cd build  
  
$ cmake ../  
  
$ make  
  
$ make test  
  
$ sudo make install
```

When the build script finishes, you will have a new directory named “BUILDS” containing a subdirectory with the build’s timestamp. An example listing of this directory follows:

Congratulations! You can now move on to installing and starting each component, as well as learning what purpose each serves.

Installation

Now that you’ve downloaded a set of official release packages or compiled your own, they need to be installed and started. A bit of background for each component will be provided, followed by the installation procedure and any initializing configuration needed. As some components depend on others, they will be presented in the order needed to satisfy these interdependencies. Change into your new build directory before continuing:

```
$ ls dev/BUILDS/2016-01-24--14-11-31/* .deb  
  
liba53_0.1_i386.deb  
libcoredumper1_1.2.1-1_i386.deb  
libcoredumper-dev_1.2.1-1_i386.deb  
openbts_5.0_i386.deb  
range-asterisk_11.7.0.4_i386.deb  
  
range-asterisk-config_5.0_all.deb  
range-configs_5.0_all.deb  
sipauthserve_5.0_i386.deb  
smqueue_5.0_i386.deb
```



```
$ cd dev/BUILDS/2016-01-24--14-11-31/
```

If you compiled your own set of packages in the previous section using the `build.sh` script, these dependencies have already been installed and this section can be skipped over. If you're using a set of official release packages, you'll need to install some additional system libraries and define an additional repository source so all dependencies can be found and installed.

Execute the following commands to define an additional repository source for ZeroMQ, a library that all the components use:

```
$ sudo apt-get install software-properties-common python-software-properties
```

```
$ sudo add-apt-repository ppa:chris-lea/zeromq
```

```
$ sudo apt-get update
```

CoreDumper library

OpenBTS uses the `coredumper` shared library to produce meaningful debugging information if OpenBTS crashes. Google originally wrote it and there are actually two `libcoredumper` packages: `libcoredumper-dev` contains development files needed to compile programs that utilize the `coredumper` library, and `libcoredumper` contains the shared library that applications load at runtime:

```
$ sudo dpkg -i libcoredumper1_1.2.1-1_amd64.deb
```

The exact version numbers found in the package names may have changed since the publishing of this book.

A5/3 library

OpenBTS uses the `A5/3` shared library to support call encryption. It contains cryptographic routines that must be distributed separately from OpenBTS:

```
$ sudo dpkg -i liba53_0.1_amd64.deb
```

Installing Components

By installing all of the following components on a fresh system, you are guaranteed a functional GSM network-in-a-box. Everything needed for voice, SMS, and data will be running in a single system.

The overall architecture of what you will be installing is visible in [Figure 5-2](#). The Session Initiation Protocol (SIP) and Real-time Transport Protocol (RTP) are the two protocols that OpenBTS uses to convert GSM traffic into VoIP.

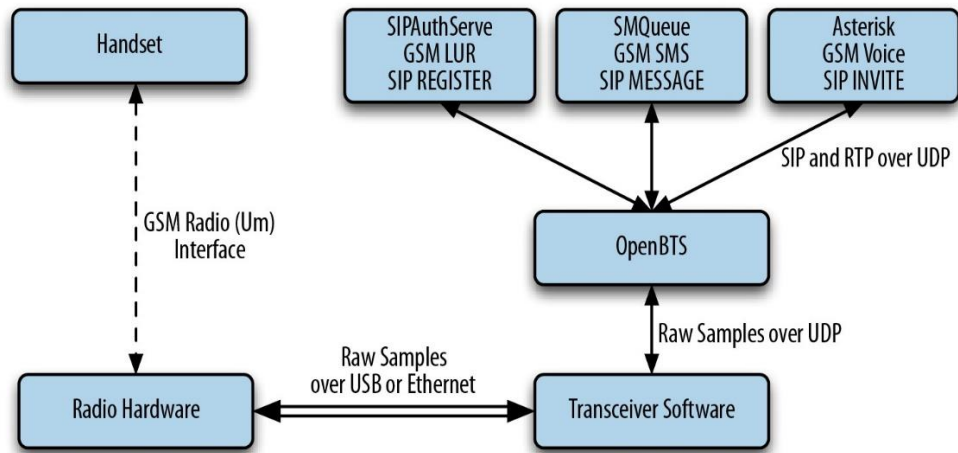


Figure 5-2: Component architecture

System configs

This package contains a set of default configurations that will allow a fresh Ubuntu system to work out of the box when installed. It includes settings for the network interface, firewall rules, domain name system (DNS) configuration, logging, etc.

You may not want to install this package if you are already comfortable configuring a Linux distribution, but its contents can serve as a guide for the required changes. During installation, you will be prompted several times to confirm the overwriting of certain

configuration files. If you are unsure what the file does, a safe answer is always “Y” when dealing with a fresh system:

```
$ sudo dpkg -i range-configs_5.0_all.deb
```

If error found related to Miss ntp or bind9 the following command will be used:

```
$ sudo apt-get install ntp
```

Asterisk

Asterisk is a VoIP switch responsible for handling SIP INVITE requests, establishing the individual legs of the call, and connecting them together. There are two packages responsible for setting up an Asterisk installation that works without any additional configuration: range-asterisk and range-asterisk-configs.

The range-asterisk package contains a confirmed-working version of the Asterisk SIP switch software and ensures that the appropriate modules needed for OpenBTS are already included. No other patches to Asterisk are included; it is simply intended to represent the latest confirmed-working version of Asterisk.

The range-asterisk-configs package contains a set of configuration files so Asterisk knows about and can communicate with the subscriber registry database. This database is where the various components store and update subscribers’ phone numbers, identities, authentications, caller IDs, and registration states. Also, by using this database, it is no longer necessary to manually edit Asterisk configuration files when adding new handsets to the network:

```
$ sudo dpkg -i range-asterisk*.deb
```

```
$ sudo apt-get install -f
```

SIPAuthServe

SIP Authorization Server (SIPAuthServe) is an application that processes SIP REGISTER requests that OpenBTS generates when a handset attempts to join the mobile network.

When a handset authenticates successfully, SIPAuthServe is responsible for updating the subscriber registry database with the IP address of the OpenBTS instance that initiated it, allowing other subscribers to call the handset:

```
$ sudo dpkg -i sipauthserve_5.0_amd64.deb
```

```
$ sudo apt-get install -f
```

SMQueue

SIP MESSAGE Queue (SMQueue) is an application that processes SIP MESSAGE requests that OpenBTS generates when a handset sends an SMS. It stores the messages, schedules them for delivery in the network, and reschedules them if the target handset is unavailable:

```
$ sudo dpkg -i smqueue_5.0_amd64.deb
```

```
$ sudo apt-get install -f
```

OpenBTS

Finally, we reach the star of the show. OpenBTS is responsible for implementing the GSM air interface in software and communicating directly with GSM handsets over it. This communication is converted into SIP and RTP on the IP network side and interacts with the components above to form the core network.

The GSM handsets see a fully compatible GSM radio access network and the core network sees standard SIP endpoints. Neither side must know that there is a layer between allowing the handsets to connect seamlessly to the IP world:

```
$ sudo dpkg -i openbts_5.0_amd64.deb
```

```
$ sudo apt-get install -f
```

Starting/Stopping Components

Now that each component has been installed, you need to start them. Components are controlled on Ubuntu with a system named Upstart. Future releases of the OpenBTS suite will support other mechanisms such as systemd, but for now, Upstart is used. To start all components, execute the following:

```
$ sudo start asterisk
```

```
$ sudo start sipauthserve
```

```
$ sudo start smqueue
```

```
$ sudo start openbts
```

Conversely, to stop all components, use:

```
$ sudo stop openbts
```

```
$ sudo stop asterisk
```

```
$ sudo stop sipauthserve
```

```
$ sudo stop smqueue
```

Each component runs in the background and will automatically restart if a fault arises. To monitor the console output for the component once it is running in the background, the following log files can be used:

```
/var/log/upstart/asterisk.log
```

```
/var/log/upstart/sipauthserve.log
```

```
/var/log/upstart/smqueue.log
```

```
/var/log/upstart/openbts.log
```

The system components have been installed and are running. The next step will be to start testing and configuring them.

Chapter 6

Initial Testing and Configuration

The software and hardware should now be in place. This chapter will guide you through some initial sanity checks, functional testing, and basic configuration customization. By the end of this chapter, you will have successfully exchanged the first SMS messages and voice calls among phones over your private mobile network.

6.1 Initial State

Some of the manual steps that follow will conflict and fail if other instances of the services are already running. To make sure that nothing else is running on this system, execute the following:

```
$ sudo stop openbts
```

```
$ sudo stop asterisk
```

```
$ sudo stop sipauthserve
```

```
$ sudo stop smqueue
```

Now you can proceed to confirm connectivity at each step in the chain before running the first basic tests.

6.2 Confirm Radio Connectivity

The first thing you should verify is that the transceiver application can communicate with the radio hardware. Different vendors have different methods for accomplishing this.

6.2.1 Ettus Research Radios

All Ettus hardware uses the Transceiver52M binary, which was installed in /OpenBTS in the last chapter. Run it as follows to see if the hardware device is detected:

```
$ cd /OpenBTS
$ sudo ./transceiver
[sudo] password for openbts:
linux; GNU C++ version 4.8.4; Boost_105400; UHD_003.010.git-119-g42a3eeb6
Using internal clock reference
-- Opening a USRP2/N-Series device...
-- Current recv frame size: 1472 bytes
-- Current send frame size: 1472 bytes
```

The example above shows a successful attempt. The transceiver can be stopped by pressing Ctrl-C. If you instead see something like the following output, there is a problem:

```
$ cd /OpenBTS
$ sudo ./transceiver
[sudo] password for openbts:
linux; GNU C++ version 4.6.3; Boost_104601; UHD_003.007.002-release
Using internal clock reference
ALERT 1745:1745 2014-09-05T22:37:00.4 UHDDevice.cpp:528:open: No UHD devices
found with address ''
ALERT 1745:1745 2014-09-05T22:37:00.4 runTransceiver.cpp:160:main:
Transceiver
exiting...
```

Ettus provides a couple of helper applications to automatically detect and inspect attached radios. Run the following command to list all attached devices. This example shows an N200 attached via Ethernet:

```
$ uhd_find_devices
-----
-- UHD Device 0
-----
Device Address:
type: usrp2
addr: 192.168.10.2
name:
serial: E9R26M7UP
```

Another helpful application is `uhd_usrp_probe`, which will inspect a device and return its technical information and configuration. An example run of this application follows:


```
$ uhd_usrp_probe
```

```
linux; GNU C++ version 4.6.3; Boost_104601; UHD_003.007.002-release  
-- Opening a USRP2/N-Series device...  
-- Current recv frame size: 1472 bytes  
-- Current send frame size: 1472 bytes
```

```
Device: USRP2 / N-Series Device
```

```
  /  
  |   Mboard: N200r4  
  |   hardware: 2576  
  |   mac-addr: XX:XX:XX:XX:XX:XX  
  |   ip-addr: 192.168.10.2  
  |   subnet: 255.255.255.255  
  |   gateway: 255.255.255.255  
  |   gpsdo: none  
  |   serial: XXXXXX  
  |   FW Version: 12.3  
  |   FPGA Version: 10.0
```

```
  |   Time sources: none, external, _external_, mimo  
  |   Clock sources: internal, external, mimo  
  |   Sensors: mimo_locked, ref_locked
```

```
  /  
  |   RX DSP: 0  
  |   Freq range: -50.000 to 50.000 Mhz
```

```
  /  
  |   RX DSP: 1  
  |   Freq range: -50.000 to 50.000 Mhz
```

```
  /  
  |   RX Dboard: A  
  |   ID: SBX (0x0054)  
  |   Serial: XXXXXX
```

```
  /  
  |   RX Frontend: 0  
  |   Name: SBXv3 RX  
  |   Antennas: TX/RX, RX2, CAL  
  |   Sensors: lo_locked  
  |   Freq range: 400.000 to 4400.000 Mhz  
  |   Gain range PGA0: 0.0 to 31.5 step 0.5 dB  
  |   Connection Type: IQ  
  |   Uses LO offset: No
```

```
  /  
  |   RX Codec: A  
  |   Name: ads62p44  
  |   Gain range digital: 0.0 to 6.0 step 0.5 dB  
  |   Gain range fine: 0.0 to 0.5 step 0.1 dB
```

```
  /  
  |   TX DSP: 0  
  |   Freq range: -250.000 to 250.000 Mhz
```

6.2.2 Troubleshooting Ethernet

Whether using a virtual machine or real server, an extra Ethernet interface must be available for your Ethernet connected radio. While it is possible to change the IP address of the radio to match your local network, this is undesirable because the samples being exchanged over Ethernet between the transceiver application and radio hardware are very sensitive to delay and loss. They should be on a dedicated connection. Install an extra physical Ethernet interface in your server or create an additional virtual Ethernet interface in the virtual machine. Make sure this interface is on the same subnet as the radio hardware. The default IP address for all Ettus hardware is 192.168.10.2. Assign an appropriate address to your extra Ethernet interface using the following example:

```
$ sudo ifconfig eth1 192.168.10.1/24
```

Now test the connection with a simple ping. Press Ctrl-C to stop the ping test:

```
$ ping 192.168.10.2
PING 192.168.10.2 (192.168.10.2) 56(84) bytes of data.
64 bytes from 192.168.10.2: icmp_req=1 ttl=64 time=1.037 ms
64 bytes from 192.168.10.2: icmp_req=2 ttl=64 time=1.113 ms
```

Please check that you are using the compatible version of UHD with USRP by using the following 2 commands:

```
$ uhd_images_downloader
```

```
$ uhd_images_loader
```

6.3 Starting Up the Network

Now that you have confirmed the transceiver software can communicate with the radio hardware, you can start running the OpenBTS service in the background. Do that with the following command:

```
$ sudo start openbts
```

The OpenBTS service will automatically start an instance of the transceiver software and connect to the radio hardware. Radio samples are then exchanged between the transceiver software and OpenBTS software over a local User Datagram Protocol (UDP) socket.

6.4 The Configuration System and CLI

All configuration of OpenBTS is accomplished by manipulating keys stored in a SQLite3 database. By default, this database is stored at `/etc/OpenBTS/OpenBTS.db`. Each key is defined in a schema that is compiled in OpenBTS and used to validate the values being used. One advantage afforded by this style of configuration system is that most key values can be changed and are applied to the running system within a few seconds without interrupting service. These are dynamic keys. There are also a few static keys in the configuration system that require a restart of OpenBTS to apply the change.

The easiest way to manipulate the configuration keys is via the OpenBTS command line interface (CLI). Run the following shell command to open it:

```
$ sudo /OpenBTS/OpenBTSCLI
```

You are now presented with an OpenBTS prompt. This is where commands, including configuration changes, can be executed for processing by OpenBTS. From now on, commands prefixed with `$` are to be executed on the Linux command line. Commands prefixed with `OpenBTS>` are for the OpenBTS command line. It may be convenient to have two terminal windows open so there is no need to constantly enter and exit the OpenBTS command line.

6.4.1 Changing the Band and ARFCN

The first things you must check are the radio band and Absolute Radio Frequency Channel Number (ARFCN) being used. The radio band is one of four values: 850, 900, 1800, or 1900 MHz, corresponding to the four GSM bands available around the world. An ARFCN is simply a pair of frequencies within the selected band that will be used for the transmission and reception of radio signals. Each radio band has over 100 different ARFCNs that can be used. ARFCN may also be referred to as the carrier (e.g., systems using multiple ARFCNs are multiple carrier systems). Choosing the correct band and ARFCN is important for regulatory reasons and to avoid interference with or from local carriers. You use the OpenBTS `config` command to inspect the current band and ARFCN settings. These configuration keys are in the `GSM.Radio` category. To view all configuration keys with the word `GSM.Radio` in their name, enter the following command:

```
OpenBTS> config GSM.Radio
GSM.Radio.ARFCNs 1      [default]
GSM.Radio.Band 900     [default]
GSM.Radio.CO 51       [default]
GSM.Radio.MaxExpectedDelaySpread 4      [default]
GSM.Radio.PowerManager.MaxAttenuation 0
GSM.Radio.PowerManager.MinAttenuation 0  [default]
GSM.Radio.RSSI Target -50    [default]
GSM.Radio.SNR Target 10     [default]
```

The `GSM.Radio.Band` key shows that the 900 MHz band is being used and the `GSM.Radio.CO` key indicates that ARFCN #51 in that band is currently selected.

If your radio hardware does not have limitations on or optimizations for a particular frequency, you can proceed with these settings. An easy optimization for eliminating interference is to choose a band that is not used by other carriers in your country. In general, the Americas use 850 and 1900 MHz systems while the rest of the world uses 900 and 1800 MHz. Also, choose a lower frequency if possible to improve coverage with lower power. If your local regulator has assigned you a specific band and ARFCN, then you must use it. To change your GSM band, you must again use the OpenBTS `config` command. This time, add the desired band to the end of the command. The following example changes the band to 850 MHz:

```
OpenBTS> config GSM.Radio.Band 850
GSM.Radio.Band changed from "900" to "850"
WARNING: GSM.Radio.CO (51) falls outside the valid range of ARFCNs 128-251
for
GSM.Radio.Band (850)
GSM.Radio.Band is static; change takes effect on restart
```

The command confirms that the band has been changed but delivers two additional pieces of information. First, there is a warning about the ARFCN not being valid any more for the 850 MHz band. The valid range is 128–251 for 850 MHz. Second, you are informed that the `GSM.Radio.Band` parameter is static and cannot be applied at runtime; OpenBTS

```
OpenBTS> config GSM.Radio.C0 166
GSM.Radio.C0 changed from "51" to "166"
GSM.Radio.C0 is static; change takes effect on restart
```

must be restarted. To fix the first warning, use the `config` command again to set a valid ARFCN for the 850 MHz band:

The command confirms that the ARFCN has been changed and warns again about this parameter being static. You can now restart OpenBTS to apply the change:

```
$ sudo stop openbts
openbts stop/waiting
$ sudo start openbts
openbts start/running, process 6075
```

The service will take a few seconds to start back up and you are again free to use the OpenBTS CLI.

6.4.2 Ettus Research Radio Calibration

One main difference between the Range Networks and Ettus Research radios is in the proper value for `GSM.Radio.RxGain`. Range Networks uses a much higher value for this parameter and if it is not adjusted, the Ettus Research equipment will not work correctly.

```
OpenBTS> devconfig GSM.Radio.RxGain 10
GSM.Radio.RxGain changed from "52" to "10"
GSM.Radio.RxGain is static; change takes effect on restart
```

The signal being received will overdrive the demodulator. For starters, set `GSM.Radio.RxGain` to 10:

There is also a dedicated command that allows you to set this parameter without restarting OpenBTS. Use `rxgain` if you need to make fine adjustments to avoid restarting each time.

6.5 Testing Radio Frequency Environment Factors

If you've never had a project that involves RF or analog signals in general, you may be surprised by the number of things that can go wrong with them. You may actually be surprised, in the end that RF communication can work at all! RF experts in the OpenBTS community are sometimes regarded as practitioners of black magic...but we digress.

In a GSM network, two separate radio frequencies are used so the base station and handsets can communicate simultaneously in both directions. Put another way, GSM uses frequency division multiple access to establish full duplex communication. The ARFCN selected is what determines which pair of frequencies will be used. The path from the base station to the handset is known as the downlink and the path from the handset back to the base station is known as the uplink.

The next thing to look out for when setting up a new network is excess radio interference or “noise” from other sources on the uplink. If the uplink is too noisy, the signals from handsets cannot reliably be demodulated into usable information. OpenBTS will show the current level by using the `noise` command:

```
OpenBTS> noise
noise RSSI is -68 dB wrt full scale
MS RSSI target is -50 dB wrt full scale
INFO: the current noise level is acceptable.
```

In this example, the detected environmental noise Received Signal Strength Indication (RSSI) is -68 dB (lower numbers are better and mean less noise is present) and the configured target RSSI level for handsets is -50 dB. This means that the base station can,

at best, receive 18 dB more energy from the handsets than the environmental noise —a very good margin meaning uplink reception issues due to noise should not be a problem.

Smaller margins between these two numbers will produce different informational messages. For example, having a margin of 10 dB or less will report:

WARNING: the current noise level is approaching the MS RSSI target, uplink connectivity will be extremely limited.

And a margin of zero or less will report:

WARNING: the current noise level exceeds the MS RSSI target, uplink connectivity will be impossible.

If either of these WARNING messages are reported, you will need to take action to reduce uplink noise and/or increase the handset transmit power.

In the future, if your handset can see the base station but can no longer connect, noise should be the first thing to check. Your configuration could still be 100% correct and functional but the radio environment may have changed, preventing communication.

6.6 Reducing Noise

If your base station radio setup does not include a frequency duplexer, the number one source of noise on the uplink can actually be the downlink signal. Without proper duplexing to filter it out, the downlink signal is usually the closest energy source to the uplink both physically and by frequency.

6.6.1 Antenna alignment

A quick duplexer of sorts is simply aligning the antennas so they do not so readily feed into each other. If you are using rubber duck–style antennas, tilt them so they form a 90 degree angle. The radiation pattern for these antennas will then be perpendicular as shown in [Figure 6-1](#).

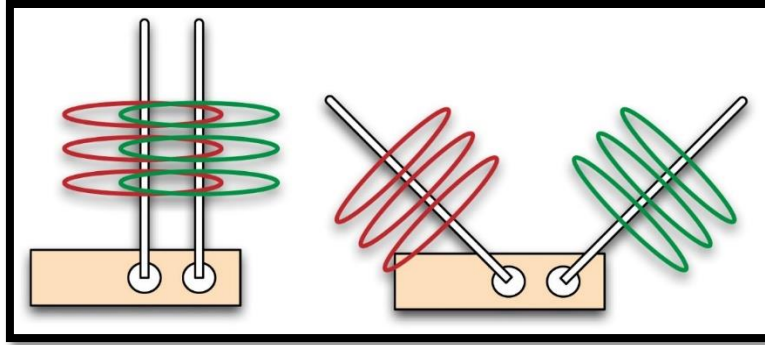


Figure 6-1: Antenna alignment

If the antennas are parallel to each other, signal can efficiently flow from the transmit to the receive antenna, but when the antennas form a 90 degree angle, the signal is being transmitted on a different plane than it is being received on. Observe the change by running the `noise` command before and after your adjustment. This simple adjustment can reduce noise by as much as 10 dB.

6.6.2 Downlink transmission power

The alignment step above reduced the flow of energy from the transmit antenna to the receive antenna. This received energy may still be too high for the uplink to be usable. Decreasing the downlink transmission power will further clean up the uplink. The coverage area lost by decreasing the downlink power is not significant in a lab environment.

Cleaner signals are preferable to strong ones. Run the `power` command with no arguments to see the current level. The power is reported in decibels of attenuation:

```
OpenBTS> power
Current downlink power 0 dB wrt full scale
```

To decrease the downlink transmission power, for example by 20 dB, enter the following:

```
OpenBTS> power 20
Current downlink power -20 dB wrt full scale
```


The downlink is now transmitting with 20 dB less power. Use the noise command to observe the improvement.

You can see all our OpenBTS configuration parameters in **Appendix B**. in order to make a good and clear phone call.

6.7 Steps to make a phone call for the first time configuration

In this section we are providing a full guide for any developer who wants to establish a phone call with the previously described installation by using USRP N210.

6.7.1 Searching for the Network

Now that the radio is calibrated and the settings are confirmed, you will use a handset to search for the newly created network. Each handset’s menu is different but the item is usually similar to “Carrier Selection” or “Network Selection.” The process for manually selecting a different carrier on Android is detailed in **Figure 6-2**.

1. Launch the “Settings” application from the Android menu system.
2. Select “More.”
3. Select “Mobile networks.”
4. Select “Network operators.” This may or may not start a search. If it does not, select “Search networks.”
5. Once the search has finished, a list of available carrier networks is presented.

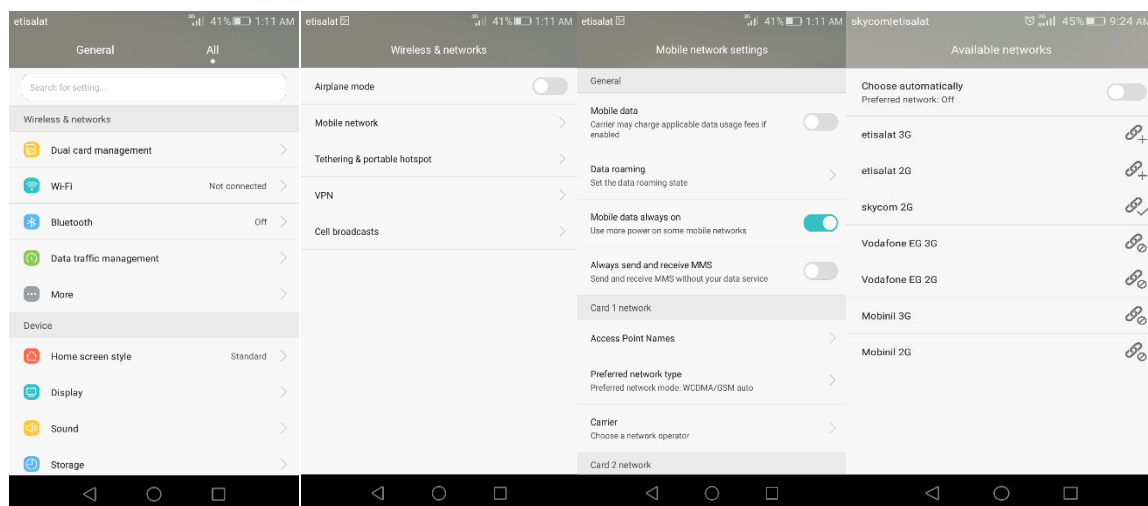


Figure 6-2: Android carrier selection

Here we see the test network in the list of selectable carriers. Depending on the handset model, firmware, and SIM used, the network ID will be displayed as “60207,” “602-07,” “Test PLMN 1-1,” or the GSM short name of “OpenBTS.” If your test network is not detected, force the search again by either reselecting the menu item, toggling airplane mode between on and off, or power cycling the handset. If that still does not work, confirm again that the handset supports the GSM band you have configured above and that the baseband is unlocked (i.e., not restricted by contract to only using a specific carrier).

6.7.2 Finding the IMSI

The main identity parameter you will be searching for is the International Mobile Subscriber Identity (IMSI). This is a 14–15 digit number stored in the SIM card and is analogous to the handset’s username on the network.

Handsets will not usually divulge the IMSI of their SIM card. It can sometimes be located in a menu or through a field test mode, but this method of determining a SIM’s IMSI is very cumbersome to explain. Luckily, there are other methods; OpenBTS also knows the IMSIs it has interacted with and, because you are in control of the network side, you also have access to this information.

To force an interaction between a handset and your test network, you will perform a location update request (LUR) operation on the network, analogous to a registration. This is nothing more complicated than selecting the network from the carrier selection list.

Before attempting any LURs, you need to start the SIPAuthServe daemon responsible for processing these requests:

```
$ sudo start sipauthserve
sipauthserve start/running, process 7017
```

Now, again following the steps in “[Searching for the Network](#)”, bring up the carrier selection list and choose your test network. After a short time, the handset should report a registration failure.

OpenBTS remembers these LUR interactions in order to perform something called IMSI/Temporary Mobile Subscriber Identity (TMSI) exchanges. IMSI/TMSI exchanges swap the user-identifiable IMSI for a TMSI and are used to increase user privacy on the network. The exchanges are disabled by default (modify `Control.LUR.SendTMSIs` to enable); however, the information is still there to inspect using the `tmsis` command. Use it now to view all recent LUR interactions with handsets:

```
OpenBTS> tmsis
```

IMSI	TMSI	IMEI	AUTH	CREATED	ACCESSED	TMSI_ASSIGNED
214057715229963	-	012546629231850	0	78s	78s	0
001010000000002	-	351771054186520	1	80h	95s	0
001010000000003	-	351771053005400	1	80h	108s	0

Entries are sorted by time, with the top entries corresponding to the most recent interactions. Your handset should be the top entry on this list—the most recent interaction with AUTH set to 0 because the LUR failed due to the handset not being a known subscriber. The other entries in this example are additional test handsets that have successfully performed an LUR as indicated by the AUTH column being set to 1.

6.7.3 Adding a Subscriber and OpenRegistration

Adding a Subscriber

You should now have all the necessary pieces of information to create a new subscriber account on your test network.

A couple of fields are still needed but are freely selectable: Name and Mobile Station International Subscriber Directory Number (MSISDN). The Name field is merely a friendly name for this subscriber so you can remember which handset or which person it is associated with. The MSISDN field is nothing more complicated than the subscriber’s

phone number. Because you are not connected to the public telephone network, this can be any number you choose.

The program you need to add subscribers is `nmcli.py`. It is a simple client for the Node Manager APIs and allows you to change configuration parameters, add subscribers, monitor activity, etc., all via JSON formatted commands.

`nmcli.py` is already present in your development directory—move there now to access it:

```
$ cd dev/NodeManager
```

There are two ways to add a subscriber using `nmcli.py`. The first creates a subscriber that will use cached authentication:

```
$ ./nmcli.py sipauthserve subscribers create name imsi msisdn
```

The second creates a subscriber that will use full authentication:

```
$ ./nmcli.py sipauthserve subscribers create name imsi msisdn ki
```

Example:

```
$ ./nmcli.py sipauthserve subscribers create "iPhone 4" IMSI214057715229963 \
6055551234
```

```
raw request: {"command":"subscribers","action":"create","fields":
```

```
{"name":"iPhone 4","imsi":"IMSI214057715229963","msisdn":"6055551234","ki":""}}
```

```
raw response: {
```

```
  "code" : 200,
```

```
  "data" : "both ok"
```

```
}
```

Perform the same command substituting your own information to add the first subscriber to your test network.

Open Registration

OpenRegistration is an OpenBTS-specific feature that provides a WiFi captive portal like implementation for mobile networks. Captive portals are familiar to anyone who has used an airport or hotel's public WiFi connection. Your device can connect to the WiFi network but is denied access to certain features until you answer a question, watch an advertisement, enter a pin, etc. The device is used to provision itself.

Similarly, OpenRegistration allows a handset to join a mobile network with initially restricted access. It may be able to dial out but the handset does not have an assigned number and as such cannot be called by other participants in the network. However, it can be used to provision its own number via SMS.

This type of network is very useful in any ad hoc installation where the users are temporary and fluid or the network itself is only temporarily needed: emergency response, remote work areas, tourist destinations, large festivals, etc. Because an administrator is not needed to create accounts and assign numbers, OpenRegistration networks are easier to deploy and still very useful for the users.

To get started with an OpenRegistration network, the feature itself must be enabled. First, take a look at the keys:

```
OpenBTS> config OpenRegistration
Control.LUR.OpenRegistration (disabled) [default]
Control.LUR.OpenRegistration.Message Welcome to the test network. Your
IMSI is [default]
Control.LUR.OpenRegistration.Reject (disabled) [default]
Control.LUR.OpenRegistration.ShortCode 101 [default]
```

To enable OpenRegistration, the `Control.LUR.OpenRegistration` key must be set to a regular expression. A regular expression (sometimes written “regex”) is a way of defining a pattern to be matched.

For this book, OpenRegistration will be enabled to accept any IMSI it encounters:

```
OpenBTS> config Control.LUR.OpenRegistration .*  
Control.LUR.OpenRegistration changed from "" to ".*"
```

6.7.4 Asterisk configurations

By reaching this step we can say that all OpenBTS configuration and all our installation are done successfully but we need to know what is asterisk and why we need to configure this application in order to make a phone call.

Asterisk is a free and open source framework for building communications applications and is sponsored by Digium. Asterisk turns an ordinary computer into a communications server. Asterisk powers IP PBX systems, VoIP gateways, and conference servers and is used by small businesses, large businesses, call centers, carriers and governments worldwide.

We need to configure this application in order to route phone call between users correctly without any problems.

The first thing we need to do is to go to this directory:

```
/etc/asterisk
```

We will find some configuration files like sip.conf, extentions-range.confetc.

extensions-range.conf editing

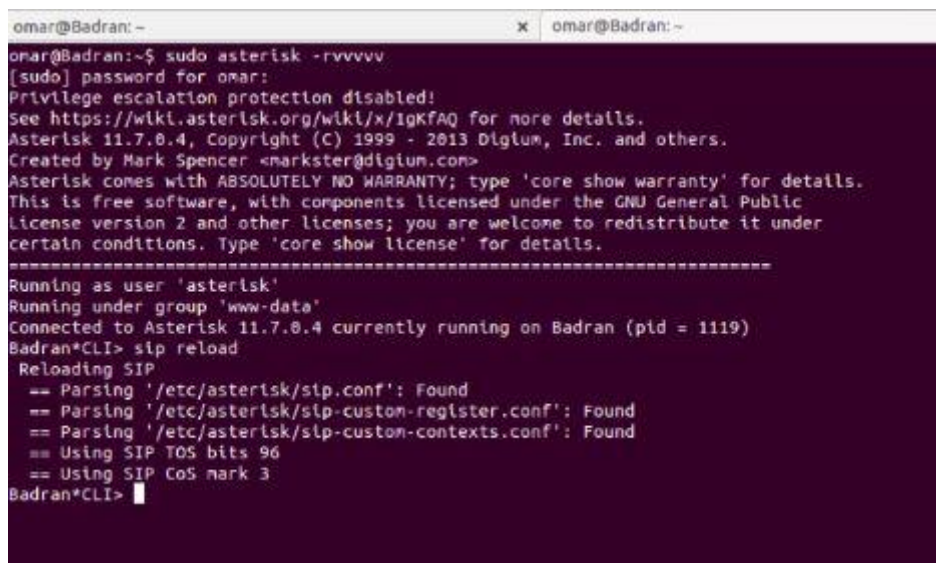
- 1- Open the text file
- 2- Type this line for each user:
exten=>phone number,1, Dial(SIP/IMSI6020300*****)

sip.conf & sip-customs-contexts.conf editing

- 1- Open the text file
- 2- Type this line for each user:
[IMSI602030*****]
callerid=01118****
canreinvite=no
host=dynamic
allow=gsm
regexten=01118****
context=phones
type=friend

After editing these files open asterisk and debug CLI by typing in the terminal as shown in [Figure 6-3](#).

```
$ sudo asterisk
$ sudo asterisk -rvvvvv
> sip reload
> dialplan reload
> sip show peers
```



```
omar@Badran: ~
x omar@Badran: ~
omar@Badran:~$ sudo asterisk -rvvvvv
[sudo] password for omar:
Privilege escalation protection disabled!
See https://wiki.asterisk.org/wiki/x/1gKfAQ for more details.
Asterisk 11.7.0.4, Copyright (C) 1999 - 2013 Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.
This is free software, with components licensed under the GNU General Public
license version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
Running as user 'asterisk'
Running under group 'www-data'
Connected to Asterisk 11.7.0.4 currently running on Badran (pid = 1119)
Badran*CLI> sip reload
Reloading SIP
== Parsing '/etc/asterisk/sip.conf': Found
== Parsing '/etc/asterisk/sip-custom-register.conf': Found
== Parsing '/etc/asterisk/sip-custom-contexts.conf': Found
== Using SIP TOS bits 96
== Using SIP CoS mark 3
Badran*CLI> █
```

Figure 6-3: Asterisk and debug CLI

Then do the following steps:

- 1) search for network
- 2) choose your network
- 3) receive SMS from 101
- 4) send your number to 101

After doing these steps you will be able to successfully make your first call and send your first SMS.

For re-registration

```
$ sudo sqliteman /var/lib/asterisk/sqlite3dir/sqlite3.db
```

Then delete all entries (dialdata , sipbuddies), After that send your number to 101 again.

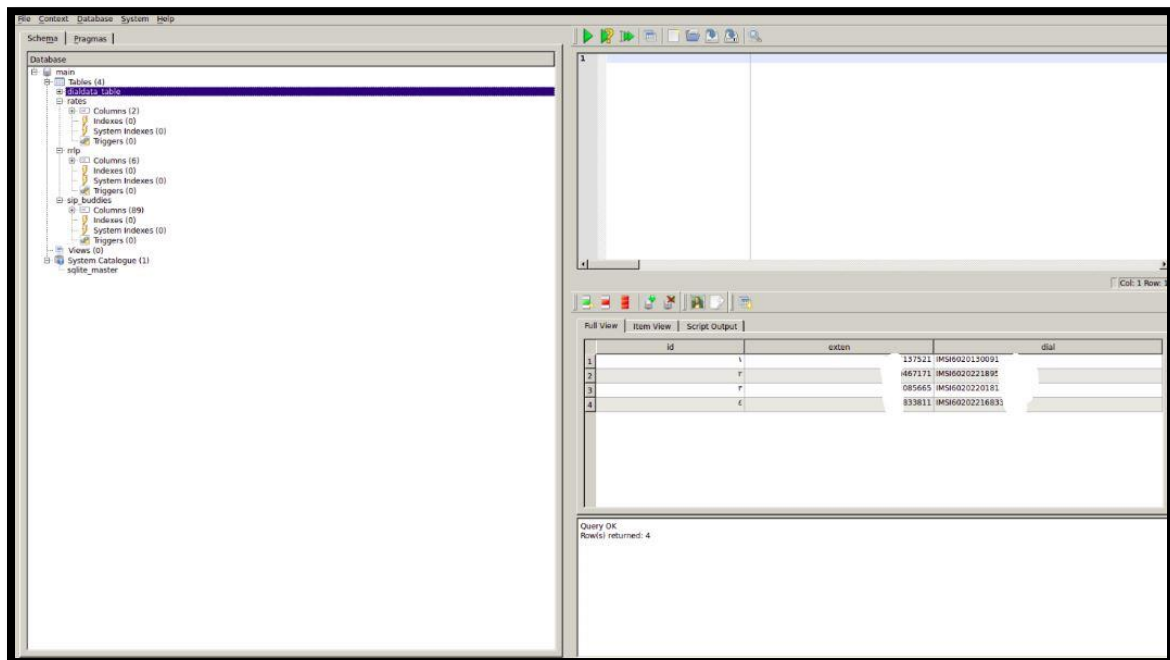


Figure 6-4: SQLite DataBase

6.7.5 First Connection

Now when you select your test network in the connection menu, the LUR should succeed. This can be confirmed with the `tmsis` command in OpenBTS. The “AUTH” column will now have a “1” in the entry corresponding to your IMSI:

```
OpenBTS> tmsis
```

IMSI	TMSI	IMEI	AUTH	CREATED	ACCESSED	TMSI_ASSIGNED
214057715229963	-	012546629231850	1	11m	56s	0
001010000000002	-	351771054186520	1	80h	8m	0
001010000000003	-	351771053005400	1	80h	9m	0

Congratulations, you’ve successfully registered to your own private mobile network! Feel free to register any additional handsets you wish to use before proceeding.

6.7.6 Test SMS

Now that a handset has access to your network, you can perform some more interesting tests. The first is a quick test of your network’s SMS capabilities.

The component responsible for receiving, routing, and scheduling the delivery of SMS messages is SMQueue. It must be started before testing out these features; execute the following command to do so:

```
$ sudo start smqueue
smqueue start/running, process 21101
```

Echo SMS (411)

On your handset, compose an SMS to the number 411. This is a “shortcode” handler in SMQueue that will simply echo back whatever it receives along with some additional information about the network and subscriber account that was used.

The body of the message to 411 can be whatever you’d like, although it can be useful to use unique content for each message or sequential numbers or letters. This helps you pinpoint which message is being responded to in case an error occurs.

Once you have your message composed to 411, hit send. After a few seconds, a reply should appear (an example follows):

This indicates the following:

```
"1 queued, cell 0.1, IMSI214057715229963, phonenum 6055551234, at Sep 8 02:30:59,  
Ping pong"
```

- There is one message queued for delivery.
- The base station has a load factor of 0.1.
- The message was received from IMSI 214057715229963, MSISDN 6055551234.
- The message was sent on September 8 at 02:30:59.
- The message body was “Ping pong.”

Direct SMS

SMS messages can also be tested directly from OpenBTS by using the `sendsms` command.

From the OpenBTS CLI, let’s see how it is invoked by using the `help` command:

```
OpenBTS> help sendsms  
sendsms IMSI src# message... -- send direct SMS to IMSI on this BTS,  
addressed  
from source number src#.
```

Messages are sent by specifying a target IMSI, the source number the message should appear to have originated from, and the message body itself. Substitute the information for your subscriber account to compose a message and press Enter:

After a few seconds, your handset should display a new incoming message from the imaginary number 8675309 with a body of “direct SMS test.”

```
OpenBTS> sendsms 214057715229963 8675309 direct SMS test  
message submitted for delivery
```

SMS messages created in this way do not route through SMQueue at all; they are sent directly out through the GSM air interface to the handset and, as such, cannot be rescheduled. If the handset is offline or unreachable, these messages are simply lost. This is why SMQueue is needed—to attempt and reschedule deliveries in the inherently unpredictable wireless environment.

Two-Party SMS

If you have configured more than one handset for use in your network, feel free to send a few messages back and forth between them.

6.7.7 Test Calls

The other service to test is voice. As with SMS, OpenBTS does not directly handle voice and requires an additional service to be run—in this case, Asterisk.

Start Asterisk now:

```
$ sudo start asterisk
asterisk start/running, process 1809
```

Using the same handset you used in the SMS tests, you will now verify a few aspects of the voice service. This is accomplished by utilizing a few test extensions that the `rangeasterisk-configs` package defines. An extension is an internal phone number, unreachable from the outside.

Test Tone Call (2602)

The first test extension you will use plays back a constant tone. This might not sound too exciting but does confirm many things about the network:

- Asterisk is running and reachable.
- Call routing is working as expected.
- Downlink audio is functional.

Call 2602 with your handset now.

As you listen to the tone, listen for changes in pitch. These changes in pitch are due to missing information in the downlink voice stream path, similar to packet loss. In the field, this is the primary use for the test tone extension: testing downlink quality. A downlink loss of 3% is normal in production networks, with losses of 5%–7% still providing an understandable conversation.

Echo Call (2600)

The next test extension creates an “echo call.” Basically, all audio that Asterisk receives will be immediately echoed back to the sender. In addition to confirming the items listed for the test tone call, the echo call will reveal any delay or uplink quality issues present in your network.

Call 2600 with your handset now. As you speak into the microphone, you should hear yourself very shortly afterward in the earpiece. A little delay is normal, but longer delays lead to an experience more like using a walkie-talkie. The human brain can deal with delays up to about 200 ms without trouble. Beyond that, the conversation starts to break down and both sides stop speaking because it becomes uncomfortable.

Two-Party Call

If you have configured more than one handset for use in your network, feel free to place some calls between them. Verify that the source numbers are correct when receiving a call.

6.8 Automatic registration code

In order to make it easy for the developer to register users in SkyComm network for the first time we have developed a C++ code which make asterisk configurations automatically only all you have to do is to run it from Ubuntu CLI. The following is the method and instructions of the code.

```
- To install c/c++ compiler
$ sudo apt-get update
$ sudo apt-get install build-essential manpages-dev

- To verify the installation
$ whereis gcc
$ which gcc
$ gcc -version

- to install some important libraries (you don't need to install it if you don't get errors
while compiling)
$ sudo apt-get install libstdc++4.8-dev:amd64
- Go to file directory
$ cd /(enter your direcotry)

- compile
$ g++ auto.c

- run
$ ./a.out imsi phoneid //you may need check the name of the program in the directory
```

Source code

```
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <assert.h>
#include <string.h>
#include <fstream>
#include <iostream>
#include <string>
#include <cstring>

using namespace std;

int main(int argc, char* argv[])
{ system("sudo chmod 777 /etc/asterisk/sip.conf");
  system("sudo chmod 777 /etc/asterisk/sip-custom-contexts.conf");
  system("sudo chmod 777 /etc/asterisk/extensions.conf");
  system("sudo chmod 777 /etc/asterisk/extensions-range.conf");
  FILE * pFile;
  pFile = fopen("/etc/asterisk/sip.conf", "a");
  fprintf(pFile, "\n[IMSI]");
  fprintf(pFile, argv[1]);
  fprintf(pFile, "\ncallerid=");
  fprintf(pFile, argv[2]);
  fprintf(pFile, "\ncanreinvite=no\nhost=dynamic\nallow=gsm\nregexten=");
  fprintf(pFile, argv[2]);
  fprintf(pFile, "\ncontext=phones\ntype=friend\n");
  fclose(pFile);
```

```
pFile = fopen("/etc/asterisk/sip-custom-contexts.conf","a");
fprintf(pFile,"\n[IMSI];
fprintf(pFile,argv[1]);
fprintf(pFile,"]\ncallerid=");
fprintf(pFile,argv[2]);
fprintf(pFile,"\ncanreininvite=no\nhost=dynamic\nallow=gsm\nregexten=");
fprintf(pFile,argv[2]);
fprintf(pFile,"\ncontext=phones\ntype=friend\n");
fclose(pFile);

pFile = fopen("/etc/asterisk/extensions.conf","a");
fprintf(pFile,"exten=>");
fprintf(pFile,argv[2]);
fprintf(pFile,",1, Dial(SIP/IMSI");
fprintf(pFile,argv[1]);
fprintf(pFile,")\n");
fclose(pFile);
return 0;
}
```

Chapter 7

Outside world

What gives a network a high value is reaching the outside world not only connecting through the inside network, the main goal of SkyComm is reaching the outside world through satellite but this was impossible due to the high cost and regulations that don't allow usage of satellite link for non-business issues. So we tried to simulate the satellite link with other techniques, some of them failed and some succeeded. We will discuss the trials in this chapter.

7.1 GPRS

Mobile networks have, in many areas of the world, been reduced to being data networks. Over-the-top (OTT) services like WhatsApp and Skype only need a data pipe and participants can connect regardless of carrier. Fees between the participants are also not dependent upon geographic location, unlike local versus long-distance charges. GPRS is much too slow to support bidirectional streaming video but can suffice for a low-quality voice call. Its speeds are ideal for email and OTT text messaging.

The world of sensors and infrastructure such as heat and flow sensors or electrical and parking meters also needs data connectivity. These low-bandwidth machine-to-machine (M2M) devices, now referred to as Internet of Things (IOT) devices, are a very common use for GPRS. GPRS is actually not a part of GSM. It was developed after GSM had been standardized and is usually referred to as 2.5G, whereas plain GSM is 2G. OpenBTS abstracts these differences and presents a unified configuration where possible.

7.1.1 General knowledge about GPRS

GPRS (general packet radio service) is a packet-based data bearer service for wireless communication services that is delivered as a network overlay for GSM, CDMA and TDMA (ANSI-I36) networks. GPRS applies a packet radio principle to transfer user data

packets in an efficient way between GSM mobile stations and external packet data networks. Packet switching is where data is split into packets that are transmitted separately and then reassembled at the receiving end. GPRS supports the world's leading packet-based Internet communication protocols, Internet protocol (IP) and X.25, a protocol that is used mainly in Europe. GPRS enables any existing IP or X.25 application to operate over a GSM cellular connection. Cellular networks with GPRS capabilities are wireless extensions of the Internet and X.25 networks.

GPRS gives almost instantaneous connection set-up and continuous connection to the Internet. GPRS users will be able to log on to an APN (Access Point Name) and have access to many services or an office network (without the need to dial-up) and remain continuously connected until they log off, only paying when data is actually transmitted. A physical end-to-end connection is not required because network resources and bandwidth are only used when data is actually transferred.

This makes extremely efficient use of available radio bandwidth. Therefore, GPRS packet-based services should cost users less than circuit-switched services since communication channels are being shared and are on a 'as-packets-are-needed' basis rather than dedicated to only one user at a time. It should also be easier to make applications available to mobile users because the faster data rate means that middle-ware currently needed to adapt applications from fixed line rates to the slower speed of wireless systems will no longer be needed.

GPRS data speeds will range from 14.4 Kbit/s (using one radio time-slot) to 115kbit/s (by amalgamating time-slots) and offer continuous connection to the Internet for mobile phone and computer users. GPRS data speeds are likely to average at about 56 Kbit/s, with between 28 and 40 Kbit/s initially. The higher data rates will allow users to take part in video conferences and interact with multimedia web sites and similar applications using mobile hand held devices as well as notebook computers.

The key drivers for operators to evolve to GPRS networks are to:

- Increase revenues by moving into the mobile data market, especially since the voice market has had profit margins squeezed with the commoditization of voice services
- Gain new subscribers who require mobile data services or do not want to invest in a PC to gain Internet access
- Retain current subscribers by offering new services
- Reduce costs due to the efficient use of network resources
- Ease of adapting applications for mobile users because high data speeds mean that middle-ware is no longer required to convert fixed applications for mobile use.

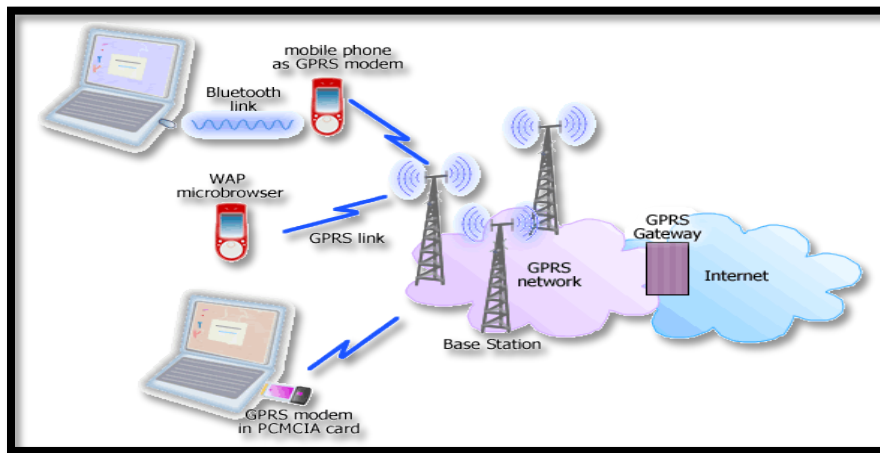


Figure 7-1: Overview of GPRS

The overall benefits of GPRS networks for mobile operators are discussed below.

GPRS is based on GSM communication and will complement existing services such as circuit switched cellular phone connections and the Short Message Service (SMS). It will also complement Bluetooth, a standard for replacing wired connections between devices with wireless radio connections.

Benefits of GPRS

From the point of view of operators:

- Offer new and improved data services to residential and business markets to aid retention and loyalty.
- Increase revenues from data services.
- Opportunity to increase subscriber numbers - there are more mobile phones in general use than there are PCs in people's homes. This means that the potential market for GPRS is high and that new Internet users are more likely to upgrade to a GPRS handset rather than making a larger investment in a PC.
- Offer innovative tariffs based on new dimensions such as the number of kilobytes or megabytes.
- Return on investment - investment in GPRS will be twofold since the new network infrastructure pieces will be used as part of the UMTS network requirements as well as GPRS.
- GPRS provides an upgrade path and test bed for UMTS.
- Control of large content portals.
- Access to the key member of the value chain – the customer
- Cost effectiveness through spectrum efficiency – with packet-switching radio resources are used only when users are actually sending or receiving data. This efficient use of scarce radio resources means that large numbers of GPRS users can potentially share the same bandwidth and be served from a single cell. GPRS spectrum efficiency means that there is less need to build in idle capacity that is only used in peak hours. GPRS therefore lets network operators maximize the use of their network resources in a dynamic and flexible way.

From the point of view of End-users:

- New data services
- Speed – higher levels of bandwidth means higher speeds for data transactions
- Cost-effectiveness – only charged when data is transmitted and not for the duration of the connection
- Constant connectivity – GPRS enables instant connections and the ability to remain logged-on at all times (Internet or corporate virtual private networks (VPN)). For example, a user with a laptop computer could be working on a document and automatically receive new e-mail which could be responded to then or later. The user has had a network connection throughout, but has not had to dial-in, as is necessary with circuit-switched connections. The immediacy of access to services is highly desirable and critical for some applications, such as remote credit card authorization where it would be unacceptable to delay customer service for several minutes
- Simultaneous voice and data communication - the user can receive incoming calls or make outgoing calls while in the midst of a data session.

7.1.2 Applying GPRS with OpenBTS

Enabling/Disabling

By default, the GPRS service is disabled in OpenBTS. Turn it on now by toggling the GPRS. Enable key:

```
OpenBTS> config GPRS.Enable 1
GPRS.Enable changed from "0" to "1"
GPRS.Enable is static; change takes effect on restart
Restart OpenBTS to apply this static key:
$ sudo stop openbts
$ sudo start openbts

Once OpenBTS has restarted, log back in to its command line and use the
GPRS list command to confirm that OpenBTS has set up a few channels for GPRS:

OpenBTS> gprs list
PDCH ARFCN=166 TN=1 FER=0%
PDCH ARFCN=166 TN=2 FER=0%
```

Central Services

GPRS does not rely on any additional components but some configuration must be in place on your Linux host for things to work correctly. This should be taken care of already from the range-configs package during setup but this is how to double-check that things are in order. The handsets' IP traffic is piped through OpenBTS and into a virtual network interface named sgsntun. You can confirm now that OpenBTS has created it by using ifconfig:

```
$ ifconfig sgsntun
sgsntun Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:500
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
```

The virtual network interface also needs routes and rules applied to it for the ip tables Linux firewall. Example rules are located in `/etc/OpenBTS/iptables.rules` and can be modified if needed to change the gateway interface name. By default, they are written for `eth0`. Apply the rules now manually:

```
$ sudo iptables-restore < /etc/OpenBTS/iptables.rules
```

To have the system apply these rules every time your `eth0` interface comes up, modify `etc/network/interfaces` to add the final line below, which contains “pre-up”:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
# The loopback network interface auto lo iface lo inet loopback
# The primary network interface auto eth0 iface eth0 inet dhcp pre-up iptables-restore <
/etc/OpenBTS/iptables.rules
```

With the tunnel device present and the rules applied, your Linux host should be in order.

Connecting

While your handset should perform an LUR and join the network once it's back up, there may be additional steps to make sure the GPRS service is recognized and usable. In GPRS this is not an LUR; it is called GPRS Attach. The handset may not perform this attach for several reasons.

If you are using another carrier's SIM card, the handset GPRS subsystem will probably consider itself to be roaming when joining your OpenBTS network. The GPRS subsystem will not attempt to Attach unless "use data roaming" has been switched on in your handset.

Also double-check your handset's Access Point Name (APN) settings. OpenBTS does not care what information is put in the name, user name, and password fields but the handset may not make an Attach request until something has been entered.

Note that: We faced a bug in our project especially in android when it comes to APN entries. In some versions, you are allowed to add a new APN but it will not save unless the Mobile Country Code (MCC) and Mobile Network Code (MNC) you enter match those defined on your SIM card. This failure is silent and no indication is given that the new APN settings will not be used. They also do not show up in the list of APNs on the device. However, if you replace the SIM card with one that matches the MCC and MNC you entered, your APN information magically reappears.

All that said, some phones may still take a few minutes to Attach. They may be presenting old network access information to OpenBTS before giving up and starting a fresh Attach. Once they have successfully attached, the handset's IP address is visible via the sgsn list command:

```
OpenBTS> sgsn list
GMM Context:imsi=001010000000009 ptmsi=0x47001 tlli=0xc0047001
state=GmmRegisteredNormal age=32 idle=0 MS#1,TLLI=c0047001,7d4373ae
TPC=192.168.99.1
```

Troubleshooting

If you are able to connect but have not received an IP address, the firewall setting may be getting in the way. Normally this is not a problem, but depending on the Linux installation and IP network configuration it is possible. Disable the firewall now and restart OpenBTS to apply the change:

```
OpenBTS> config GGSN.Firewall.Enable 0
GGSN.Firewall.Enable changed from "1" to "0"
GGSN.Firewall.Enable is static; change takes effect on restart
```

OpenBTS also tries to detect your DNS server settings and pass them on to the handsets. If you find the handset unable to resolve domain names, try setting a DNS server manually and restart OpenBTS to apply the change:

```
OpenBTS> config GGSN.DNS 8.8.8.8
GGSN.DNS changed from "" to "8.8.8.8"
GGSN.DNS is static; change takes effect on restart
```

Performance Tuning

OpenBTS attempts to intelligently divide resources between standalone dedicated control channels (SDCCHs) for signaling and TCH for media. However, it does not yet have a mechanism to balance different types of TCH usage.

Voice versus GPRS

Both voice and GPRS traffic use time slots that carry TCH logical channels. If your network will be deployed to primarily serve GPRS instead of voice, you should adjust the GPRS.Channels.Min.C0 key that specifies the minimum number of available TCH time slots that should be used for GPRS. The default value is two. To view the current number of available channels, use the load command:

```
OpenBTS> load
== GSM ==
SDCCH load/available: 0/4
TCH/F load/available: 0/5
PCH load: active, total: 0,0
AGCH load: active, pending: 0, 0
== GPRS ==
current PDCHs: 2
utilization: 0%
```


Here we see that there are seven total TCH channels available: five for GSM voice (GSM: TCH/F) and two available for GPRS data (GPRS: PDCHs). To maximize the network for GPRS, set GPRS.Channels.Min.C0 to seven, the total number of TCH channels available, and restart OpenBTS to apply the change:

```
OpenBTS> config GPRS.Channels.Min.C0 7
GPRS.Channels.Min.C0 changed from "2" to "7"
GPRS.Channels.Min.C0 is static; change takes effect on restart
```

Rerunning load will show that all TCH channels have been assigned to GPRS:

```
OpenBTS> load
== GSM ==
SDCCH load/available: 0/4
TCH/F load/available: 0/0
PCH load: active: 0 total: 0
AGCH load: active: 0 () pending: 0
== GPRS ==
current PDCHs: 7
utilization: 0%
```

OpenBTS allows you to change all time slot assignments if you'd like to control them manually. You can adjust how many Combination 1 (TCH) versus Combination 7 (SDCCH) time slots are assigned. You can also adjust the ordering of those time slots, and where the GPRS time slots should appear in multi-ARFCN systems. Take a look at all the keys by searching for "Channels":

```
OpenBTS> config Channels
GPRS.Channels.Min.C0 2 [default]
GPRS.Channels.Min.CN 0 [default]
GSM.Channels.C1sFirst 0 [default]
GSM.Channels.NumC1s auto [default]
```

Individual Handset Throughput

Handsets can use more than one time slot concurrently to access GPRS. The number of concurrent time slots that are supported for uplink and down link is known as a multi slot class. By default, OpenBTS is set to support a 3+2 multi slot class: three concurrent time slots for down link and two for uplink. This does not mean that one handset will dominate all five time slots concurrently, but you may now see why adding as many time slots as possible to a GPRS-focused network is important. A higher multi slot class will deliver higher speed data to a single handset, but the network will become more quickly congested as multiple handsets attempt to use it.

You can adjust the supported multi slot class using the following two keys:

```
OpenBTS> config Multislot
GPRS.Multislot.Max.Downlink 3 [default]
GPRS.Multislot.Max.Uplink 2 [default]
```

Coverage Area versus Throughput

GPRS defines four different coding schemes (CSs) for the data being delivered. They are appropriately named CS1, CS2, CS3, and CS4. Coding Scheme 1 has the lowest throughput but the highest reliability. It assigns more bits to be used for backing out errors introduced during the radio transmission. Coding Scheme 4, on the other hand, has the highest throughput but is susceptible to errors during transmission. These transmission errors limit the usable coverage area for each coding scheme.

OpenBTS supports CS1 and CS4. You can choose between low throughput but a reliable and, thusly, larger coverage area, or higher throughput with less robust error correction, resulting in a smaller coverage area. By default, OpenBTS has both enabled, but they can be adjusted individually on the up link and down link:

```
OpenBTS> devconfig Codecs
GPRS.Codecs.Downlink 1,4 [default]
GPRS.Codecs.Uplink 1,4 [default]
```

Why did we fail?

We were able to connect to the GPRS as an up link only, we had problems with the down link and after we have done some research we have found that there is no synchronization between the frequency of GPRS in mobile equipment and the frequency the GPRS is received. This can be hardly solved by writing a special code to detect this frequency but it will take us along time and it could be a new whole graduation project.

After many trails on GPRS, we didn't succeed so we had to find another method to go to the outside world as it is one of the main targets in our thesis. Therefore we started working in UMTS.

7.2 UMTS

7.2.1 General knowledge about UMTS

Universal Mobile Telecommunication System (UMTS) is envisioned as the successor to GSM. UMTS also addresses the growing demand of mobile and Internet applications for new capacity in the overcrowded mobile communications sky. The new network increases transmission speed to 2 Mbps per mobile user and establishes a global roaming standard.

UMTS, also referred to as wide-band code division multiple access (W-CDMA), is one of the most significant advances in the evolution of telecommunications into 3G networks. UMTS allows many more applications to be introduced to a worldwide base of users and provides a vital link between today's multiple GSM systems and the ultimate single worldwide standard for all mobile telecommunications.

International Mobile Telecommunications–2000 (IMT– 2000)

The main characteristics of 3G systems, known collectively as IMT–2000, are a single family of compatible standards that have the following characteristics:

- Support both packet-switched (PS) and circuit-switched (CS) data transmission.
- Offer high data rates up to 2 Mbps (depending on mobility/velocity).

- Offer high spectrum efficiency.
- ITU-TT is a set of requirements defined by the International Telecommunications Union (ITU).
- The most important ITU-TT proposals are the UMTS (W-CDMA) as the successor to GSM.
- UMTS is being developed by Third-Generation Partnership Project (3GPP) which found in December 1998 to test and continue development of UMTS.

More advantages of 3G

- Improved speech quality.
- User-friendliness.
- World-wide access.
- World-wide HPLMN services.
- Specific service definition.
- Fast transfer of large data.
- (Inter-/Intra-net, File Transfer, E-Mail, Multimedia).

UMTS networks are now to be designed on the basis of the existing GSM infrastructure and are to be downward compatible with GSM. UMTS has a modular design for this reason.

Evolution data transmission

In UMTS, UTRA introduces a new multiple access method (WCDMA), modulation principle (QPSK) and a 25 times larger bandwidth than GSM at new frequency ranges. New RAN network elements and protocols are defined. The maximum data transmission rate will be some 2 Mbit/s.

7.2.2 Why did UMTS fail and GSM work in SkyComm?

The target of our project is to allow the passenger to make calls during his flight by his own SIM card... so our network has to register the user in database to be able to accept calls request or to deliver calls, the most important data is the IMSI of the user's SIM card.

In 2G family technology the network need to authenticate that the sim card is compatible with the network identity, if it is compatible the network will accept the user but if not the network will find that the sim card is for another network then the access will be denied.

Authentication in 2G technology: it is one way authentication that means that the network has to check the user but the user doesn't have the right to check the network.

The authentication of a user depend on an important parameter which is the secret key (Ki),it is a number implemented by hardware in the user's SIM card which not allowed to be sent in a wireless way for security issues and this number is stored in the database of the network.

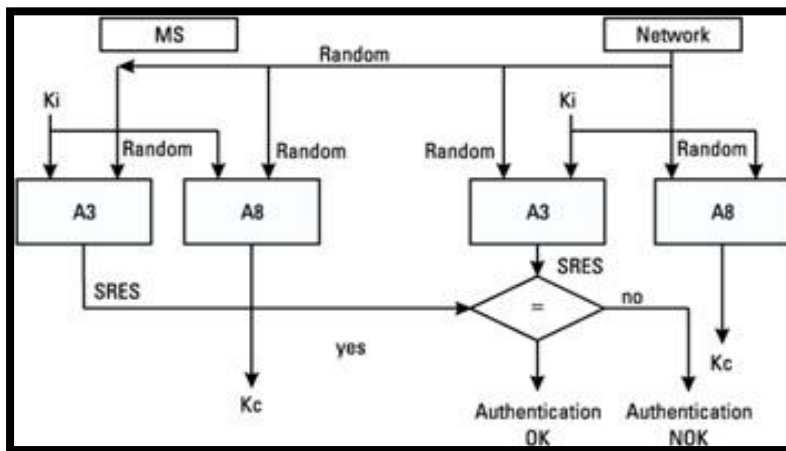


Figure 7-2: Authentication process in 2G technology

As shown in **Figure 7-2** To register the network the system compare between the secret key (ki) of the user with the stored secret key (ki) in the data base, this authentication starts from the network by creating 128 bit randomly called random number and use a function called A3 which is near to xoring function to get an output called response 1, mobile equipment of the user will receive the random number from the network and use the A3

function too, then reply to the network with the output of this function which called response 2.

Network will compare response 1 with response 2, if equal the user will be registered, otherwise the user will not registered. The above figure shows the process of 2G authentication.

In 2G technology, although our database doesn't have the user secret key (ki), it will register the user by accepting any response send by user. So any user can be added to the network easily.

In UMTS family technology

The network can't accept any user. This is because the authentication is 2 way authentications, so the user's sim card has to make sure first form the network before sending its IMSI then allow the network to make sure from it.

Network will not only send random number to the user, it will send the random number and the AUTN (Authentication taken) which contains output bits of functions with inputs depends mainly on the secret key (ki) which registered in the network. This way is complicated and we don't have to explain the process in details but the full process is shown in the figure below which called authentication vector (AV). As shown below

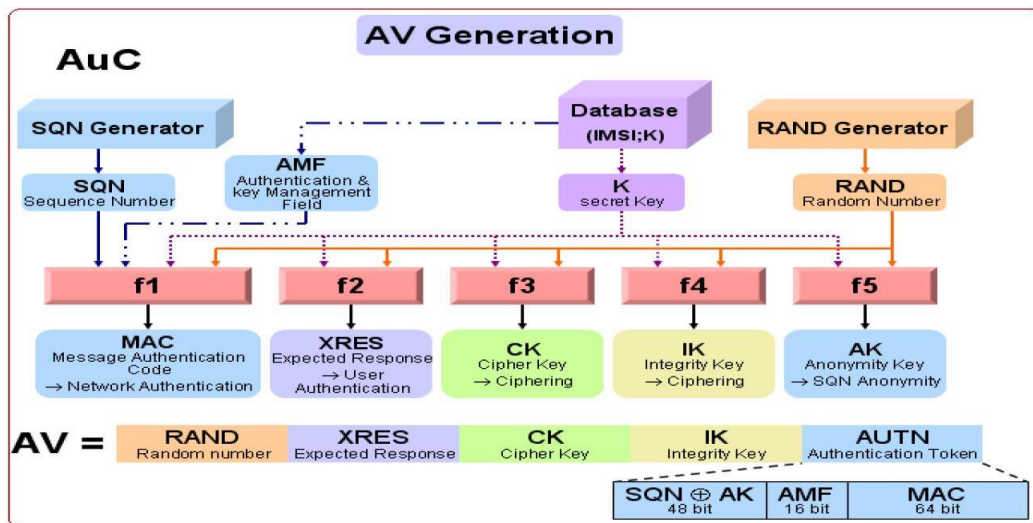


Figure 7-3: Authentication vector generation

The user SIM card will analyze a part from the received AUTN and get parameter will be used in some functions, one of those functions will produce another parameter must equal the other part of the received AUTN.

If and only if the user's SIM card authentication is successfully done, user will send response to the network which is an output of another function. The network will compare this response with its response which depends also on the secret key (ki), and then the network will add the user to its database, the full authentication process in sim card is shown in figure below.

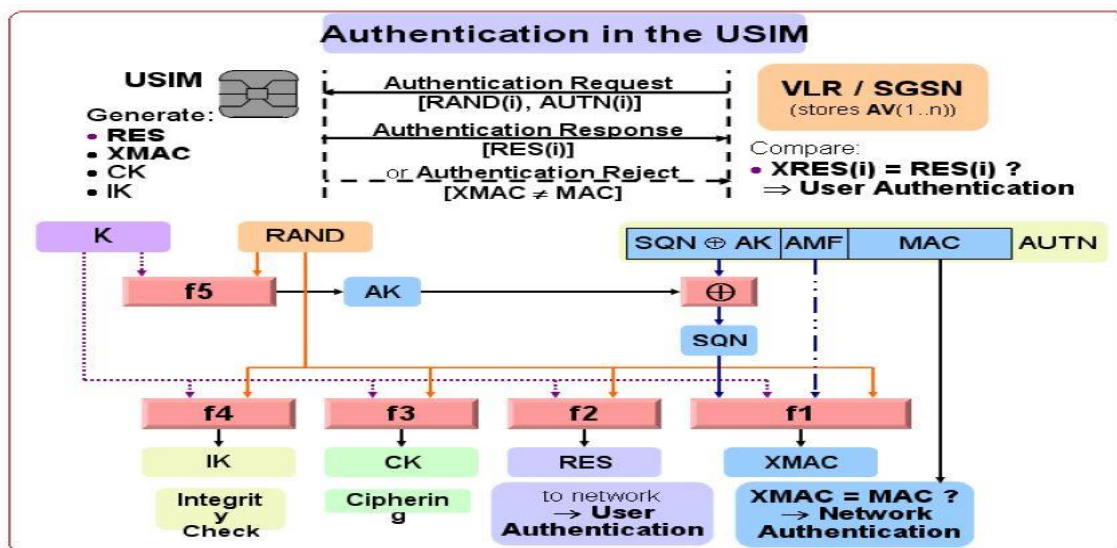


Figure 7-4: Authentication in user SIM

In our project, it is impossible to apply UMTS because we don't have the secret key of the users SIM card, the only way to get the secret key (ki) is by making a roaming agreement with the users' operators to be able to get the user data. No operator will accept to give small company or students such important data about their users. The operator will give those data to get a gain form the other company which must be authenticated too.

So the only way to apply UMTS in our project is using blank SIM cards with known secret key (ki) then we will build our database, but unfortunately this is not the target of our project which mentioned above.

This problem was mentioned on the OpenBTS official website as the subscriber registry will need to know the secret key (ki) to platform authentication and enable integrity protection, without the authentication and integrity protection, the UE will not register to OpenBTS-UMTS. For most users, this means you must provide the SIMs for the UEs on the network. The only way to use SIMs from another provider is to obtain the secret key (ki) through a roaming interface to the provider HLR.

As this method had many obstacles and was extremely complex to be implemented due to the above reasons we had to try another trail which is VOIP.

7.3 Voice over IP (VOIP)

What is VoIP?

Voice over Internet Protocol (VoIP) is a form of communication that allows you to make phone calls over a broadband Internet connection instead of typical analog telephone lines. Basic VoIP access usually allows you to call others who are also receiving calls over the Internet. Interconnected VoIP services also allow you to make and receive calls to and from traditional land line numbers, usually for a service fee. Some VoIP services require a computer or a dedicated VoIP phone, while others allow you to use your land line phone to place VoIP calls through a special adapter.

VoIP is becoming an attractive communications option for consumers. Given the trend towards lower fees for basic broadband service and the brisk adoption of even faster Internet offerings, VoIP usage should only gain popularity with time. However, as VoIP usage increases, so will the potential threats to the typical user. While VoIP vulnerabilities are typically similar to the ones users face on the Internet, new threats, scams, and attacks unique to IP telephony are now emerging.

VoIP techniques

Unlike in the circuit-switched telephony world, where the technology used for the transmission and interconnection of telephony services was defined many years ago and is implemented on a national basis, VoIP operators have several possibilities at the moment of choosing the VoIP technique that will be implemented in their networks. So far, there is not a widely accepted standard for the provisioning of VoIP services. Standardization bodies such as the International Telecommunications Union (ITU) and the Internet Engineering Task Force (IETF) have defined VoIP architectures and protocols that have been deployed during the last two decades.

A specific VoIP system will be used, depending on the business strategy of the voice operator. The importance of the definition of the VoIP technology that is implemented lies in the fact that LRIC cost models are based on efficient network architectures. This chapter describes a few of the most well-known technologies used by VoIP operators for the provisioning of voice services, as well as the relevance of VoIP traffic and quality of service in cost modeling of VoIP services.

Why we failed to apply this method?

As it is illegal in Egypt, according to the regulations of NTRA. The port of transferring the call is closed due to security. In addition VOIP is much cheaper than the operators so it would make an economic crisis.

Due to the Illegal and security issues and according to our engineering ethics we decided not working with this methods and after research we found another method which is the VOIP switch.

7.4 Twinkle the succeeded technique

Twinkle is a free and open source software application for Voice over Internet Protocol (VoIP) voice communications in IP networks and instant messaging communication using SIP protocol. It is designed for GNU/Linux operating systems and uses the QT toolkit for its graphical user interface. It also features direct IP-to-IP calls. Media streams are transmitted via the Real-time Transport Protocol (RTP) which may be encrypted with the Secure Real-time Transport Protocol (SRTP) and the ZRTP security protocols.

The main advantages of Twinkle that It's an open protocol (unlike a closed protocol like Skype), so any program that supports SIP rather than being tied to a specific client so it will be easier to connect it with asterisk and it supports GSM calls with 13 kbit/s payload, 8 kHz sampling rate.

Some features of Twinkle in addition to making basic voice calls and messages:

- 2 call lines
- Multiple active call identities
- Custom ring tones or Mute
- Call Waiting and hold
- 3-way conference calling
- Call redirection on demand, when busy and no answer.
- Call reject
- History of call detail records for incoming, outgoing, successful and missed calls
- Simple address book
- Support for UDP and TCP as transport for SIP
- Secure voice communication
- MD5 digest authentication support for all SIP requests
- Identity hiding

7.4.1 Installation Guide

It will be easy to install `twinkle` while using Ubuntu 12 but if you are using Ubuntu 14.04 LTS like this project, you will find a trick. In Ubuntu 14.04 LTS, `Twinkle` soft-phone application is somehow packaged wrongly so that due to missing dependencies, `Twinkle` cannot start up in graphical mode. Until a corrected package is released, we will resurrect `Twinkle` by using the version from Ubuntu 12.04 LTS by the following steps:

1. Add the Ubuntu 12.04 package sources to the system by adding

```
Deb http://archive.ubuntu.com/ubuntu/
```

```
to /etc/apt/sources.list:
```

2. Add the following lines to `/etc/apt/preferences.d/preferences` (create the file if it does not exist):

```
Package: *
```

```
Pin: release a=precise
```

```
Pin-Priority: 400
```

Install `twinkle` from Ubuntu 12.04 LTS Precise Pangolin by issuing the following commands:

```
$ apt-get update
```

```
$ apt-get install twinkle/precise
```

This forces to install the `twinkle` package from of Ubuntu 12.04. If you had installed `twinkle` before from Ubuntu 14.04, `apt-get` will warn you that you are going to downgrade the package and will show the current version. It might also install some further libraries from Ubuntu 12.04. `Twinkle` should be started again. It is now the older version which works without problems and has the correct dependencies so that the graphical interface comes up.

3. After having installed the old version, you have to fix it by command:

```
$ apt-get upgrade
```

To make sure that will not overwrite with the broken version from 14.04 again, add another entry to `/etc/apt/preferences.d/preferences`:

```
Package: twinkle
```

```
Pin: version 1:1.4.2-2.1
```

```
Pin-Priority: 500
```

Note that the needed pin is the version number of twinkle which appeared after installing in case of overwriting.

4. Finally, `apt-get` might complain about some auto-installed packages which are not needed any more. These are libraries needed by 14.04's twinkle which are not needed by any other package. Remove these with

```
$ apt-get autoremove
```

Now we have a working twinkle. But unfortunately, after installing twinkle this way, some people said that it will become more and more unstable after about 3 months, probably due to updated libraries or other stuff, which moved to the system more and more away from precise environment.

7.4.2 Configuration Guide

To start twinkle, use the following command:

```
$ sudo twinkle
```

The following steps show how to configure twinkle with asterisk:

- 1) The first step to work with twinkle is making an account by either editor or wizard, this is shown in the [figure 7-5](#) which shows the main window of twinkle GUI.
- 2) The most important information must be added while creating profile which can be edited also by choosing Edit form the previous figure is the user name and the domain while the other user data have less priority shown in [figure 7-6](#). We must

refer that if more than one laptop is working with twinkle, one of them must act as server to other slaves. So the domain of all twinkles must be the IP address of the server. So all hosts must either the server or other slaves must be connected as the same network. But by assigning real IP to the server, the slaves could reach the server from anywhere as its IP is unique.

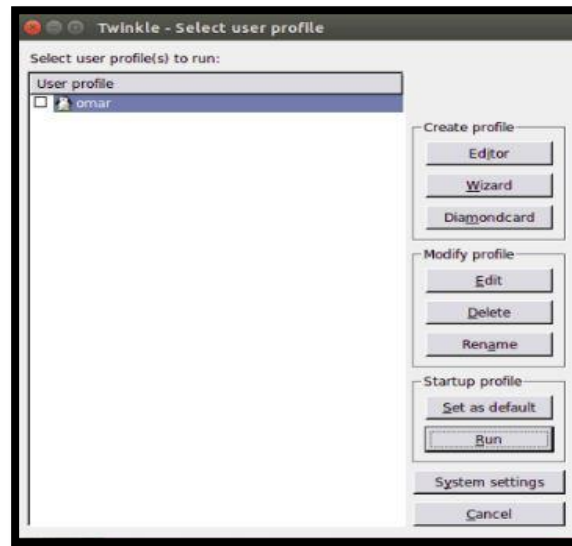


Figure 7-5: Startup window of Twinkle GUI

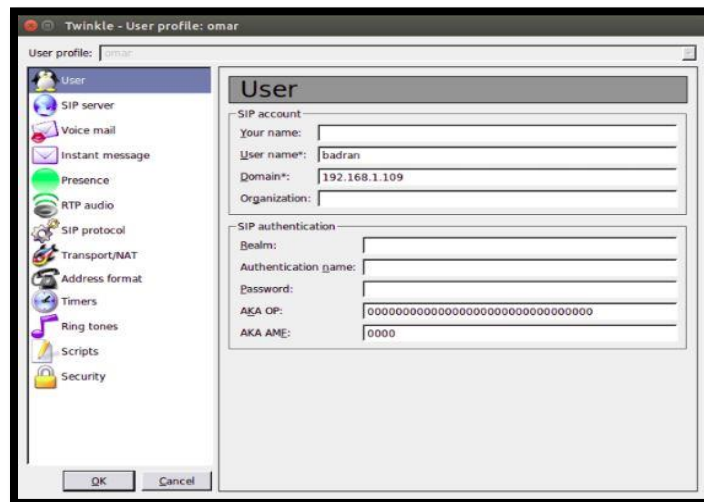


Figure 7-6: Twinkle user profile

- 3) Sip port of twinkle must be edited before starting connection with asterisk with number which is not interface with other used port numbers. Port number can be edited from system settings as the following figure shows.

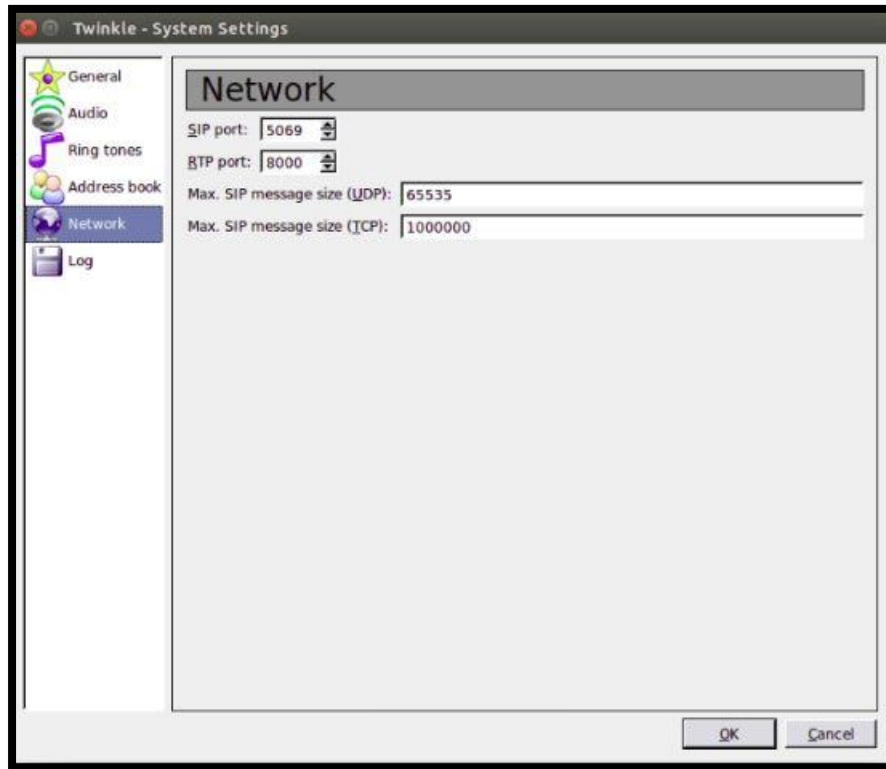


Figure 7-7: Twinkle system settings

- 4) The last step is to connect with asterisk, some data must be as to the file configurations of asterisk with the same way which is explained in chapter 6 but by replacing IMSI number with the user name of twinkle user and by assigning random caller id with any number of digits as 1234, 567 or any other numbers of digits.
- 5) When you run twinkle profile, it will display that the registration succeeded. So it's ready to connect to other twinkles or mobiles connected to SkyComm network. So you can enter any registered number to asterisk to the call label or receive calls from GSM or soft-phone as the following figure shows.

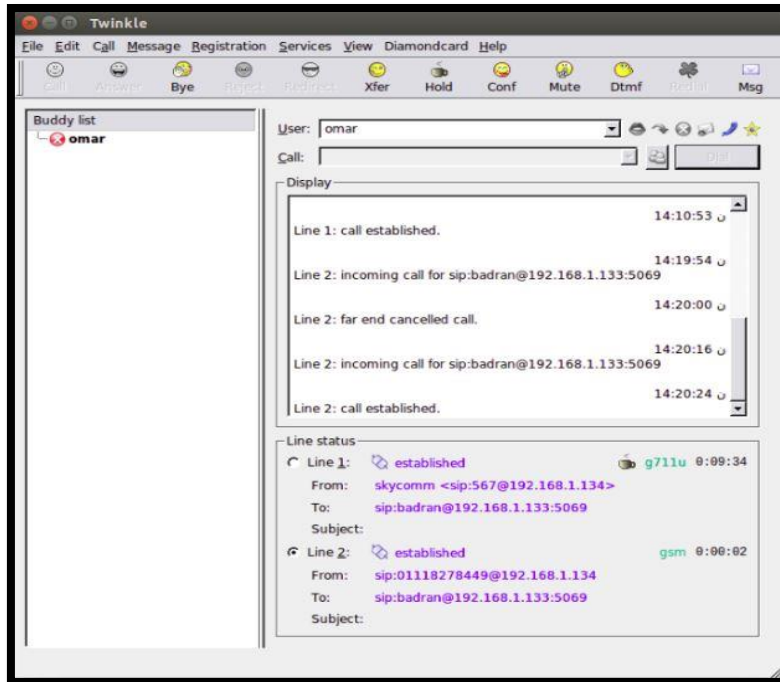


Figure 7-8: Established calls with GSM and softphone on twinkle

Note: To close twinkle correctly and avoid error of used SIP port, twinkle must be closed from the little star which found at the top left of the following figure and maybe found anywhere too.

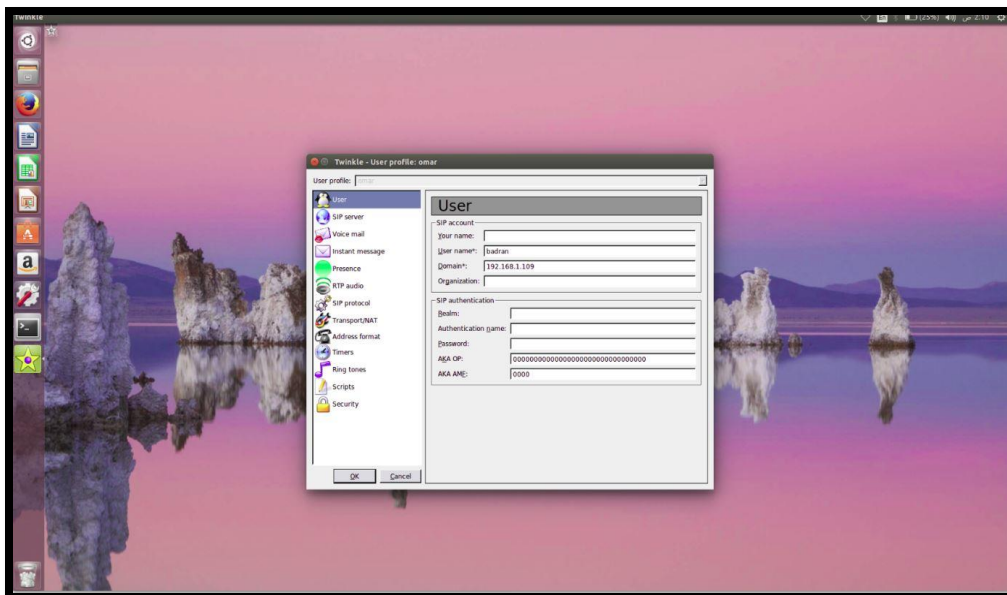


Figure 7-9: Close Twinkle correctly from star

Chapter 8

Business case model

8.1 Motivation

Marketing for engineering-focused companies is often split into two disciplines: technical marketing and ordinary marketing. Functionally, technical marketing deals with the internal workings parts and how engineers can use them to solve real-world problems. As the name implies, this is a hybrid discipline that requires technical (engineering) excellence and at least rudimentary skills in more general marketing. Ordinary marketing is often focused on finding ways to get people to identify themselves as potential users (buyers) of products.

So we need to be more focus in our field to provide an optimum solution but we should know about some statistic about the companies that support calling service in planes (Technical marketing).

8.2 Statistics & Research

Keeping customers connected to mobile networks in-flight would be a major opportunity for U.S. carriers -- potentially worth \$2.4 billion plus a year, according to Akshay Sharma, a wireless network analyst at Gartner.

Now that the federal government is considering an end to its in-flight phone call ban, these but it will cost cell phone companies millions of dollars to install the proper equipment on planes, so analysts expect carriers to recoup those costs with a per-flight fee similar to how in-flight Wi-Fi is used today. Wireless carriers could also charge hefty per-minute voice fees and roaming charges could apply if your cell phone company's network isn't supported on your flight. Companies might finally have a chance to dip into untapped potential revenue.

"It's a huge coup for the telecom carriers; this opens up a massive market for them," said Ari Zoldan, CEO of communication technology firm Quantum Networks.

Zoldan's company would be among those retrofitting jets with satellite technology. To keep calls crisp and uninterrupted, the plane would need to be connected constantly, even as it travels at 39,000 feet going 550 miles per hour. That would mean installing a large, powerful, computer-like device that can transmit signals to satellites in space and antennae on the ground.

Phone calls from planes equipped with technology from OnAir, a Swiss company that counts British Airways, Emirates and Singapore Airlines among its customers, is about \$3 to \$4 per minute, said Aurélie Branchereau-Giles, the director of communications for the company. *“You would expect similar pricing”* in the U.S.

Passengers using OnAir’s service, as well as AeroMobile, another provider, connect to the cellular network like it’s an international network, and are later billed through their wireless carriers.

Many of the people who took to social media to complain about the possibility of having a neighbor barking into a cell phone during a flight would no doubt welcome prohibitively high service costs. As the Journal reports, 51 percent of people have negative feelings about passengers making phone calls on planes, according to a survey cited by the FAA in an advisory group report from earlier this year.

But experts say that in part due to the likely high expense of making a call longer than a few minutes, any fear of having to endure someone else’s flight-long phone call is overblown. A picocell system would likely require customers to pay roaming fees for each minute that they are talking on the phone, much like how international roaming is charged today. Virgin Atlantic’s AeroMobile service is only available to customers of British carriers O2 and Vodafone and costs £1 per minute for calls and 20 pence for text messages, for example (it’s also limited to six users at a time).

Therefore we want to provide many solutions with market aspect by showing prices of all components of our project.

8.3 System pricing

For any new communication system there is a cost we should calculate it and then decide if we will apply this system or not. So we are going to talk about the price of every component in the system and the number of kits required.

The main components of the system:

- 1- USRP kit
- 2- RF daughterboard
- 3- Antenna
- 4- Cables
- 5- Satellite Data unit

8.3.1 USRP kit

Table 2-1: list of available USRPs Prices

Model	ARFC N	Interface	Operating frequencies	DAC	ADC	Price
N200	1	Gigabit Ethernet	DC-6 GHz	16-bit, 400 MS/s*	14-bit, 100 MS/s	1,775 \$
B200	1	USB 3.0	70 MHz –6 GHz	14-bit, 128 MS/s	12-bit, 64 MS/s	790 \$
2900	1	USB 3.0	70 MHz –6 GHz	12-bit, 61.44 MS/s	12-bit, 61.44 MS/s	1,095 \$
2901	2	USB 3.0	70 MHz –6 GHz	12-bit, 61.44 MS/s	12-bit, 61.44 MS/s	1,600 \$
E100	1	Embedded	DC-6 GHz	14-bit, 128 MS/s	12-bit, 64 MS/s	1,300 \$
X300	2	Gigabit Ethernet	DC-6 GHz	16-bit, 800 MS/s	14-bit, 200 MS/s	4,550 \$

8.3.2 RF daughterboard

There are many modes of daughterboard such as receiving mode only or transmitting mode only or full duplex mode.

Table 8-2: list of available Daughterboard prices

Model	Type	Frequency Range	Bandwidth (MHz)	Power output (mW)	Price
LFTX	Rx	0-30 MHz	60	n/a	90 \$
Basic RX	Rx	1-250 MHz	100	n/a	90 \$
TVRX2	Rx	50-860 MHz	10	n/a	240 \$
DBSRX2	Rx	800 MHz – 2.35 GHz	60	n/a	180 \$
LFTX	Tx	0-30 MHz	60	1	90 \$
Basic TX	Tx	1-250 MHz	100	1	90 \$
WBX	Full-Duplex	50 MHz – 2.2 GHz	40	100	565 \$
SBX	Full-Duplex	400 MHz – 4 GHz	40	100	565 \$
RFX900	Full-Duplex	750-1050 MHz	30	200	300 \$
RFX1800	Full-Duplex	1.5 GHz – 2.1 GHz	30	100	300 \$
RFX2400	Full-Duplex	2.3 GHz – 2.9 GHz	30	50	300 \$

8.3.3 Antennas

There are many types of antennas can be used as we can use directional or omnidirectional antenna. Also we can extend the coverage using leaky feeder.

Table 8-3: list of available antennas prices

Model	Antenna type	Price
LP0410	directional antenna	46 \$
LP0965	directional antenna	46 \$
VERT400	omni-directional vertical antenna	46 \$
VERT900	Omni-directional vertical antenna	36 \$

8.3.4 Cable

There 2 types of cables to connect the USRP kit to the processing unit:

Table 8-4: list of cables prices

Model	Length	price
USB Cable	1.8M	5 \$
1 Gigabit Ethernet cable	3M	10 \$

8.4 System Capacity

To calculate the capacity (number of Kits used in the system), firstly we have to know:

- 1- The aircraft dimensions and number of passengers.
- 2- Erlang B table.

8.4.1 The aircraft dimensions and number of passengers

Airbus A380-800 -as shown in the figures below -is the world's largest passenger airliner –we make our calculations depending on it–, and the airports at which it operates have upgraded facilities to accommodate it. And it provides seating for 525 people in a typical three-class configuration or up to 853 people in an all-economy class configuration. And there is another aspect we need to know the dimensions of the aircraft (length=72.7m and width=79.7).



Figure 8-1: Airbus A380-800

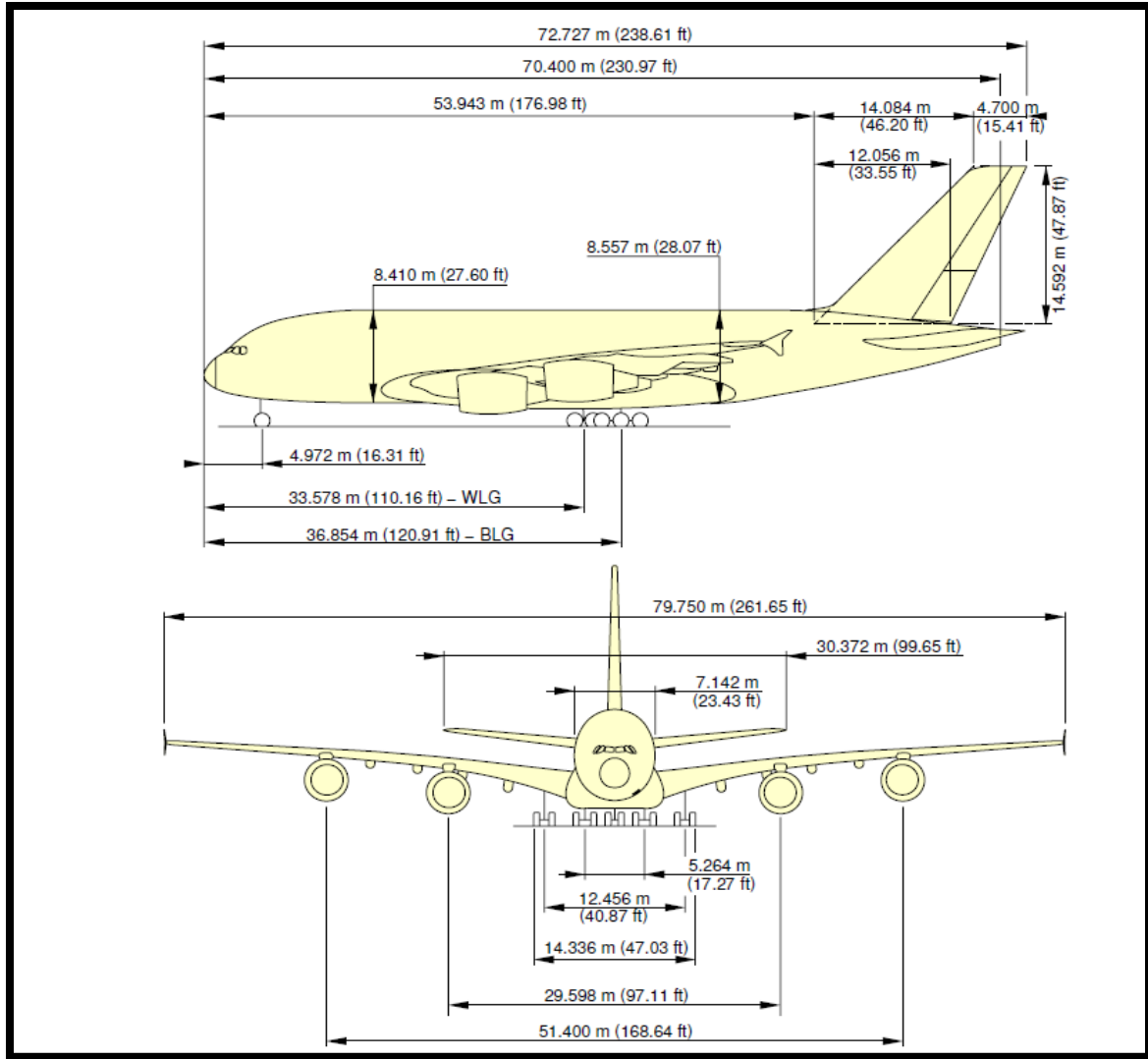


Figure 8-2: Airbus A380 Dimensions

8.4.2 Erlang B table

The erlang (symbol E) is a dimensionless unit that is used in telephony as a measure of offered load or carried load on service-providing elements such as telephone circuits or telephone switching equipment.

It depends on:

- 1- Blocking probability: it is the fraction of time a call request is denied because all channels are busy. This probability is usually specified for a given system. It is typically desired to be 3%.
- 2- Number of ARFCN

By determining the number of ARFCN and blocking probability, we can calculate how many erlang the system can support in which we can determine the system capacity. You will find Erlang B tables at appendix C.

8.5 Determining number of ARFCNs (USRP kits)

After knowing the dimensions and number of passengers of the largest aircraft and using the erlang tables we have found that if we take only in our consideration the dimensions of the plane (length= 72m) we will need only 3 kits as the coverage radius of VERT900= 15m, but if we take on our consideration the number of passengers on the plane, we should determine the erlang per subscriber –with blocking probability= 3%– to know the suitable number of kits which illustrated on table 8.5.

We take two cases to determine number of kits depending on number of erlangs per subscribers:

1st case: each subscriber has 0.01 E

2nd case: each subscriber has 0.05 E

Table 8-5: number of kits and number of subscribers with fixed Erlang

Number of kits (ARFCNs)	Channels (TCH/F)	Erlangs (3% blocking)	Subscribers (0.01 E/sub)	Subscribers (0.05 E/sub)
1	7	3.25	325	65
2	14	8.80	880	176
3	21	14.9	1490	298
4	28	21.2	2120	424
5	35	27.7	2770	554

In the first case we can have only 2 kits but due to the length (75 m) we should have 3 kits but in the second case the least number of kits we can have is 5 according to the number of subscribers(number of passengers onboard).

We have discussed before in this chapter the solution using USRP kit and its components there are another methods that provide the solution like picocells & femtocells and also there are used by airlines companies.

Chapter 9

FemtoCell vs. USRP

As a cellular standard evolves, there are many variables that must be optimized. A few of these variables are directly related to higher throughput, increased user capacity, and in general improved system performance. The optimization techniques have involved higher order modulation, improved multiple access (in both time and frequency), more powerful error correction codes, cell size reduction, and more. We believe the trend of reducing the inter-cell site distance is the correct solution leading to the femtocell.

In this chapter we are giving a brief overview about femtocell as it can be used in our solution instead of USRP and providing the reason why we are couldn't test it in our project.

9.1 FemtoCell Design

What is a femtocell? In the broadest sense, we can use the following definition: a femtocell is: a low-power base station communicating in a licensed spectrum, offering improved indoor coverage with increased performance; functioning with the operator's approval; offering improved voice and broadband services in a low-cost, technology-agnostic form factor. Here we have purposely stressed specific key descriptions to convey our message. With the intention of operating indoors, the femtocell will transmit with low power in an authorized frequency band. One of the many benefits of operating in an authorized frequency band is that the operator has the sole rights to utilize it. Hence, the operator controls who communicates in that band and can guarantee a certain level of QoS to all who are involved in occupying the private band.

More specifically, the femtocell concept entails using a low-power base station; a cellular phone; and broadband Internet access such as XDSL, cable, or fiber-to-the-home (FTTH). In the residential case all traffic would be routed through the home's ISP connection. This

concept is used not only to extend and provide cellular service but also to encourage other applications. The femtocell is sometimes called a personal base station (PBS) or Home NodeB (as referred to in the 3GPP standards body).

9.1.1 The Femtocell Concept

Year upon year cellular service providers struggle to plan for subscriber growth. In order to be prepared for this inevitability, service providers analyze various cell site deployment options. In heavily congested areas the solution has followed a theme to reduce the inter-site distance and provide micro- and even pico-cellular service.

While providing superior system quality of service (QoS) performance, improving cellular coverage is absolutely pertinent, although it can be a daunting task when one tries to satisfy not only the outdoor and highly mobile user but also the indoor and leisurely mobile user. The wireless user will encounter a vastly different experience due to the physical nature of the propagation phenomenon.

It is well known that the lower frequency bands have better propagation characteristics than the higher frequencies and will allow signals to penetrate buildings to reach the indoor users. Moreover, the lower frequency bands improve the link budget, thus allowing the use of higher-order modulation, lower processing gains, etc., which results in higher data throughput to the user. This is part of the reason for the almost absolute about-face from the technology providers racing toward the higher frequency bands to their attempting to revive the lower-frequency bands such as 450 MHz and 700 MHz.

The Femto user is still accessible by the cellular service provider but has freed up resources in the public macrocell that can now be used by additional users that are physically located outdoors. In doing so, the service provider must allow access into their private core network to provide the capability of sending user traffic to the home. This access is provided in the form of a gateway, specifically a femtocell gateway. This provides a dual benefit. First the network operator can now alleviate a fraction of their backhaul traffic to the ISP network. This freed-up capacity will be easily consumed by new users entering the network. The second benefit is to the end user—a higher data rate link can now be established to your phone. Now here is where it gets exciting: a higher data rate will ignite an influx of creative applications to be written for target cell phones.

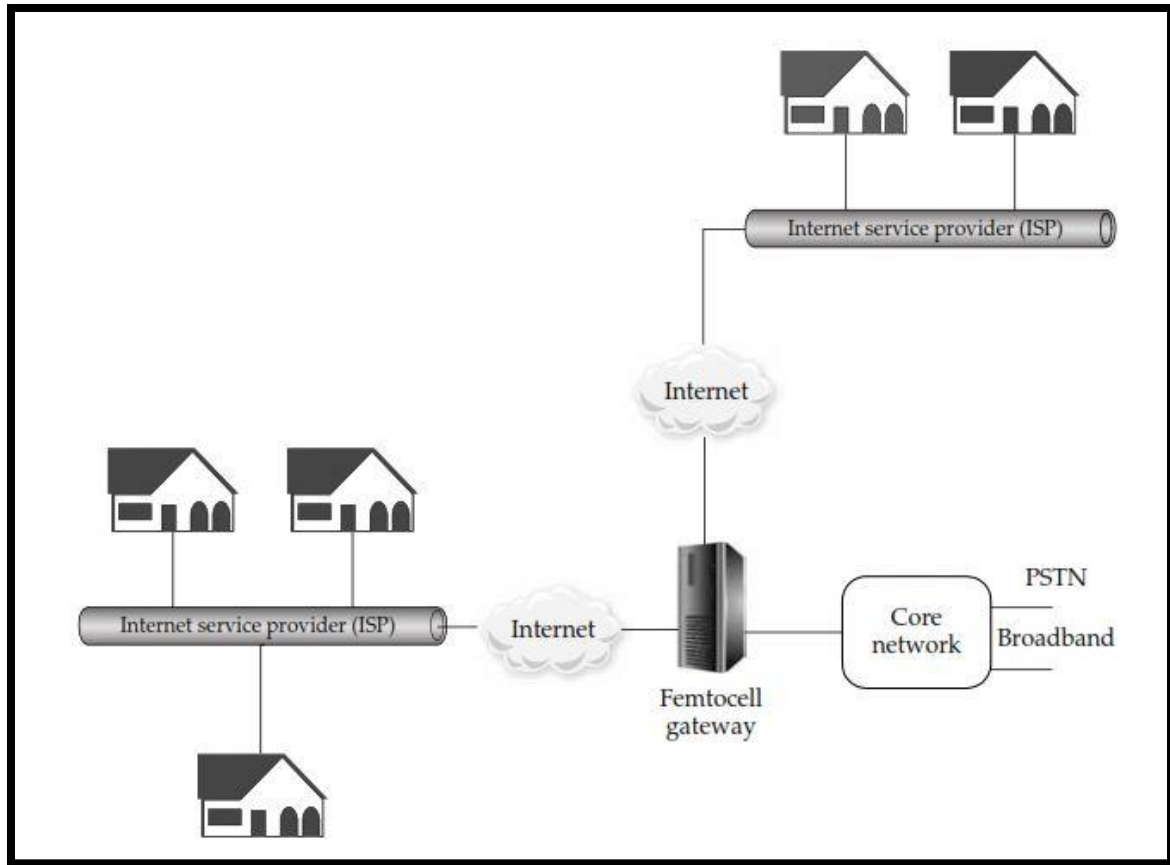


Figure 9-1: Architecture overview of a femtocell network

In **Figure 9-1** we show a sample network overview of the femtocell deployment. The homes are expected to have a broadband modem connection (i.e., XDSL, cable, or fiber) through their Internet service provider (ISP) to the Internet. The cellular specific data will be funneled through the femtocell and enter the femtocell (mobile) gateway for access back into the cellular network. For the ThirdGeneration Partnership Program (3GPP) network, the gateway would interface to the core network; this interface is called Iu-h.

9.1.2 Typical Deployment

In this subsection we will outline the items to consider for a typical deployment.

The femtocells are allowed to transmit in certain frequency bands. Service providers have purchased large chunks of frequency bands for large sums of money. These frequency

bands are referred to as licensed bands. This operational scenario is very different from the well-known WLAN case, which happens to use the unlicensed bands (ISM bands of 2.4 GHz and 5 GHz, etc.).

It is commonly accepted that a significant portion of cellular phone calls are started from within buildings. Hence we should consider cases when phone calls (or traffic sessions) initiate indoors and then eventually may need to be carried over to the outdoor public macro-network. In this case a hard handoff is made from one cell to another. Similar reasoning can be applied in the opposite direction, where phone calls were originated outdoors and then enter the femtocell coverage area.

Increasing the number of macrocells in a network is expensive. Pushing part of the network build-out to the end customer can alleviate some of the operator costs; however, for a deployment to be considered successful, millions of customers should be using the service.

How Will the User Configure the Femtocell?

After turning on the femtocell, do we rely solely on the GPS signal to obtain the femtocell geographic coordinates to be transmitted back to the cellular network to identify if this femtocell is indeed in a valid area? This may very well be the case, but it is worth mentioning that the access IP address and/or cellular neighbor cell list can also be used to help with HNB authentication and registration. We believe a combination of the above or others will lead to an accurate indication of location.

Main types of femtocell:

1-domestic: 4 voice calls and more phones connect it in standby form.

2-enterprise: larger device, from 8 to 32 call, in populated environments.

3-metro femto: operators set a large number of femto cells in an area to be less solution than building a BTS. And used for 4G LTE technology.

9.2 Femtocell or USRP?

In this part of chapter we will make a small comparison between the two solutions.

The Role of Femtocell and OpenBTS

Femtocell improves the coverage and capacity, coverage is improved because it can fill in the gaps and eliminate loss of signal through building, and capacity is improved by reduce the number of users that the operator resources serve and replace the operation using the internet network till reach the operator's infrastructure.

On the other hand OpenBTS is not used for the same reason as it is not designed to improve the coverage and capacity, but it's found to replace whole the mobile communication system and make a movable mobile network that apply mobile signal in the not converged areas.

The Price of each device

Femtocell price starts from 40\$ up to 250\$ while the USRP can reach 3000\$ approximately. We can check all part of USRP in chapter 8.

The Market aspects in each solution

Femtocell is easier in market, it is sold or loaned by a mobile network operator (MNO) to the customer. It's in the typical size of DSL or smaller and easy to use as it's a plug and play device and no specific installation or technical knowledge required. Once plugged in, it's connects to the MNO's network. The user must allow his mobile equipment to connect the femtocell just once at a time.

OpenBTS solution is more complicated. Although the code is open source and anyone can use it, the USRP device is not cheap and, allowed in some countries and the configuration is not flowed easily.

Why we didn't use femtocell solution although it's much easier and cheaper?

Because the mobile operators don't activate this solution in Egypt as they are the relay between customer and femtocell. In some countries they activate this solution and also make many promotions for customers. We wish to activate this service which can solve many problems in coverage with low costs.

Chapter 10

Conclusion

To wrap up things, Software Defined Radio has been an interesting and concerning topic to a lot of researchers and engineers, with the exponential growth in the ways and means by which people need to communicate, data communication, voice communication, and video communication. In addition it modifies the radio devices easily and lowers the cost which has become one of the most critical point in business. It also brings flexibility and less power consumption. In this bachelor thesis we used these capabilities of the Software Defined Radio to implement our communication system during flight. This project progress has mainly passed through four main major parts.

Firstly, we started to get acquainted and familiarized with the SDR by reading its main components and block diagrams from outside and decided to see which of the components would suit us. We mainly chose the N210 USRP as our platform, it was chosen according to many measures which was discussed previously. Moreover we chose also WBX as our daughter board as it supported the frequency of GSM and it was the available one for us. After that we started to read and get familiar with the Linux Ubuntu commands and how to deal with it. Adding to that, we have read about the frequencies used in the navigation system of the plane to assure safety and not to allow any interference and we found there is no safety issues or problems that would happen. Moreover, the length of the plane and maximum users to know the number of USRPs kits we need and to know the coverage area we will need. Moreover we decided to use the GSM at 900 MHZ due to an accurate study discussed before. After becoming more knowledgeable about the tools, technologies and the USRPN210 we started the second step.

The second step was reading and making a deep research about the SkyComm solution. Which was divided into many blocks. One of these blocks was the FemtoCell which was replaced by our platform the USRP and after making a small comparison between the two platforms we found that the FemtoCell is the cheapest method but we mainly chose the

USRP to learn more about this platform, adding to that the FemtoCell was not available. Another block that was used was the leaky feeder, which is an adaptation of the standard coaxial cable, with one key difference; the outer conductor is slotted and punctured allowing the cable to radiate. These cables are installed above the ceiling panels along the whole aircraft at very low radiation power level. The third block was the ANC, it generates a broadband noise floor which is being emitted through existing leaky line antenna masking reception, they measure and ensure that handsets can only connect to on board GSM network and will then operate with the lowest possible transmission power level GSM-1800 power control level its nominal output power of 0 dbm, which is used for safety issues on the plane. Adding to that the satellite unit which is a ready installed device in the aircraft which is responsible for air/ground communication this part we had to study that is it feasible to connect the USRP to this block and we found that we are able to connect easily to it. Afterwards the call will go in a full duplex way from satellite to earth station and vice versa. We were afraid from a problem which was the Doppler effect during call but after research we found that no Doppler effect will happen as the USRP is in the plane has the same speed of plane and from the point of view of the satellite it is already used for the navigation of the plane so no problem for this part. After getting deep in the solution we had to apply what we have learned and searched in real and try to simulate the solution in real so we started the third phase.

The third phase was installing the system on the USRP N210 and making our internal network by making two mobiles communicate with each other but both must be connected to the USRP which was the hard part as there was a very scarce sources about the information and documentation. We started in implementing the GSM Base station using SDR kit and GNU radio. Throughout the implementation process, we have learned how to alter some GSM parameters such as the ARFCN, allocation of phone numbers, adding mobile to the network and frequency which was 900 MHZ. This phase was completely a success and we reached our goal which is covering about 15 m radius in the lab which was not like the theoretical measure 25 m as there was many obstacles in the lab and we were allowed to perform a call, sending and receiving an SMS between two mobiles but both mobiles must be connected to the same USRP, as said before that both mobiles must be connected to the same network. In this part we were able to make 3 calls

simultaneously and non-limited number in sending a message. In addition, we have developed a code to auto register a new user without changing it manually in our configuration files. After passing this phase in a complete success we had to make the final phase.

The last phase was simulating the satellite link as we called it the outside world we were not able to buy or rent a satellite link as it is very expensive and was not available to us. So we had to get to the outside world. We tried four methods, three failed and one succeeded. The first trail was the GPRS which was the longest part as we were able to add the GPRS dependencies, libraries and codes but it didn't work. After long search we found the following that there was no synchronization at the receiver this could be solved by developing a code to track and synchronize but it was out of our focus of our thesis. So, after the failure of this idea we went to the UMTS solution but it was not supported and not applicable due to the two way security and secret KI. Therefore we went to a third solution which was the VOIP solution especially SKYPE we didn't have a lot of trials in this method as it is illegal in Egypt according to the NTRA regulation and adding to that they closed the port responsible for transferring calls through VOIP. So finally but not last the suitable and successful method was the Twinkle it is a soft phone on the laptop that could make a call between two laptops but both laptops must be in the same Local area network. Simply, this program can communicate with the Asterisk and vice versa so it was the suitable method to simulate the satellite link or as we said the outside world. This phase was a complete success for us also.

Finally, we hoped to go further and apply it on a real plane but facilities was not available for us. Moreover we wished to change it into a commercial product in many aviation companies as it will be an invading technique of communication in aviation industry and would return a large profit on any company that adopts this project.

References

- [1] Vasco Pereira and Tiago Sousa, “*Evolution of Mobile Communications: from 1G to 4G*,” IEEE Communications Magazine, July. 2004.
- [2] Tachikawa, Keiji, “*A perspective on the Evolution of Mobile Communications*”, IEEE Communications Magazine, October 2003, pp. 66-73.
- [3] “*OFDM implementation in communication system using USRP N210*”, first Edition, GUC, June 4, 2013.
- [4] “*Communication System Using USRP N210*”, first Edition, GUC, June 4, 2013.
- [5] Range Networks, Inc., “*OpenBTS Application Suite. User Manual*,” April, 15. 2014.
- [6] Leaky feeder. Retrieved from: https://en.wikipedia.org/wiki/Leaky_feeder
- [7] W.L. Gore & Associates, Inc. “*Leaky Feeder Antennas for Airborne Wi-Fi*,” Microwave Journal, October 15, 2013.
- [8] Leaky feeder cable. Retrieved from:
<http://www.solwise.co.uk/downloads/files/leaky-feeder-cable-introduction.pdf>
- [9] Dennis Roddy, *Satellite communication*, 3rd Edition. McGraw-Hill, 2001.
- [10] M.Werner, M.Holzbock, “*Aeronautical broad band communication via satellite*,”
- [11] Ground Station. Retrieved from: https://en.wikipedia.org/wiki/Ground_station
- [12] J.C. Palais, “*Fiber Optic Communications*” Fifth Edition, Prentice Hall, 2005
- [13] Michael Iadema, Range Networks, Inc., “*Getting Started with OpenBTS*,”
- [14] <http://www.twinklephone.com/>
- [15] <https://www.ettus.com/product/>
- [16] [https://en.wikipedia.org/wiki/Erlang_\(unit\)](https://en.wikipedia.org/wiki/Erlang_(unit))

Appendix A: Gaussian Minimum Shift Keying (GMSK)

Gaussian Minimum Shift Keying (GMSK): is a form of continuous-phase FSK in which the phase change is changed between symbols to provide a constant envelope. Consequently it is a popular alternative to QPSK.

The RF bandwidth of this modulation technique is controlled by the Gaussian low-pass filter. The degree of filtering is expressed by multiplying the filter 3dB bandwidth (B) by the bit period of the transmission (T), i.e. by BT (BT = 0.3 for GSM networks).

A.1 GMSK Basics

GMSK modulation is based on MSK, which is itself a form of continuous-phase frequency-shift keying. One of the problems with standard forms of PSK is that sidebands extend out from the carrier. To overcome this, MSK and its derivative GMSK can be used to reduce sideband power, which in turn reduces out-of-band interference between signal carriers in adjacent frequency channels.

MSK and also GMSK modulation are what is known as a continuous phase scheme. Here there are no phase discontinuities because the frequency changes occur at the carrier zero crossing points. This arises as a result of the unique factor of MSK that the frequency difference between the logical one and logical zero states is always equal to half the data rate. This can be expressed in terms of the modulation index, and it is always equal to 0.5.

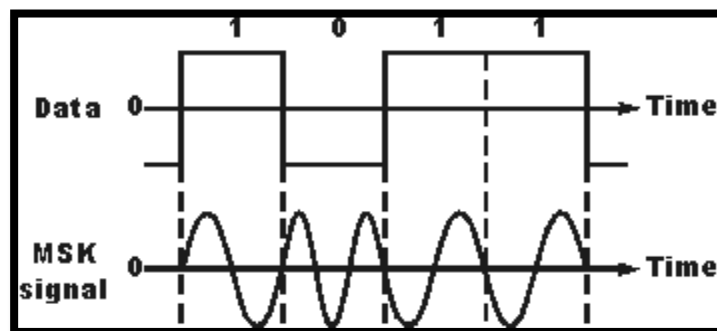


Figure A-1: Signal using MSK modulation

A plot of the spectrum of an MSK signal shows sidebands extending well beyond a bandwidth equal to the data rate. This can be reduced by passing the modulating signal through a low pass filter prior to applying it to the carrier. The requirements for the filter are that it should have a sharp cut-off, narrow bandwidth. The ideal filter is known as a Gaussian filter which has a Gaussian shaped response to an impulse. In this way the basic MSK signal is converted to GMSK modulation.

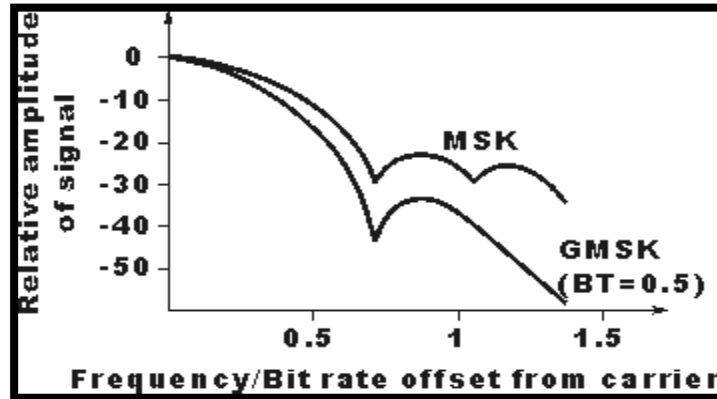


Figure A-2: Spectral density of MSK and GMSK signals

A.2 GMSK modulation

There are two main ways in which GMSK modulation can be generated. The most obvious way is to filter the modulating signal using a Gaussian filter and then apply this to a frequency modulator where the modulation index is set to 0.5. This method is very simple and straightforward but it has the drawback that the modulation index must exactly equal 0.5. In practice this analogue method is not suitable because component tolerances drift and cannot be set exactly.

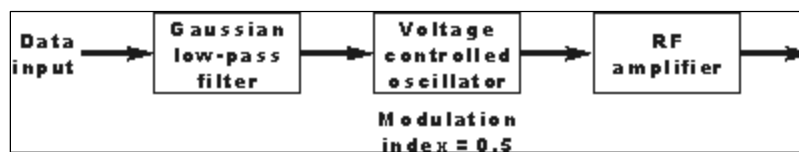


Figure A-3: Generating GMSK using a Gaussian filter and VCO

A second method is more widely used. Here what is known as a quadrature modulator is used. The term quadrature means that the phase of a signal is in quadrature or 90 degrees to another one. The quadrature modulator uses one signal that is said to be in-phase and

another that is in quadrature to this. In view of the in-phase and quadrature elements this type of modulator is often said to be an I-Q modulator. Using this type of modulator the modulation index can be maintained at exactly 0.5 without the need for any settings or adjustments. This makes it much easier to use, and capable of providing the required level of performance without the need for adjustments.

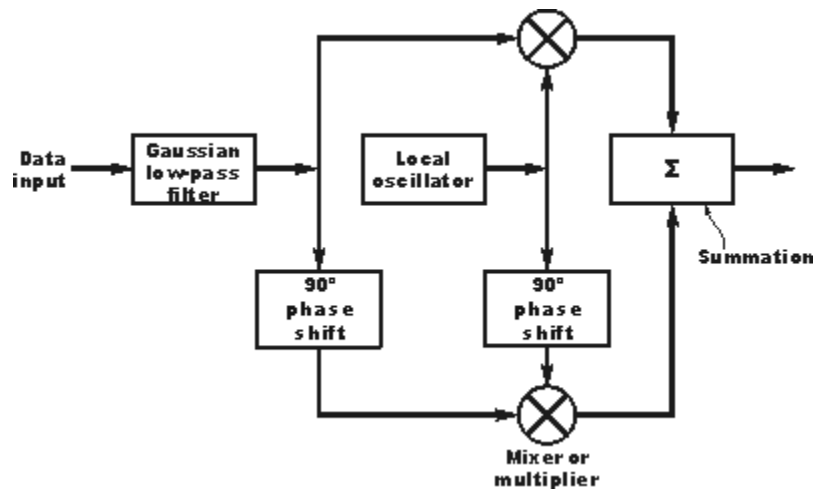


Figure A-4: Block diagram of I-Q modulator used to create GMSK

A.3 Advantages of GMSK

- 1) High spectral efficiency.
- 2) Reducing sideband power.
- 3) Excellent power efficiency due to constant envelope.
- 4) Good choice for voice modulation.
- 5) 5- Use class c nonlinear amplifiers which means that the power consumption for a given output is much less, and this results in lower levels of battery consumption; a very important factor for cell phones.

A.4 Disadvantages of GMSK

- 1) the Gaussian filter increases the modulation memory in the system and causes inter symbol interference (ISI), making it more difficult to discriminate between different transmitted data values and requiring more complex channel equalization algorithms such as an adaptive equalizer at the receiver.
- 2) Higher power level than QPSK.

Appendix B: OpenBTS configurations

B.1 All OpenBTS parameters configurations

```
Control.GSMTAP.GPRS 0 [default]
Control.GSMTAP.GSM 0 [default]
Control.GSMTAP.TargetIP 127.0.0.1 [default]
Control.LUR.404RejectCause 0x04 [default]
Control.LUR.AttachDetach 1 [default]
Control.LUR.FailMode ACCEPT [default]
Control.LUR.FailedRegistration.Message Your handset is not
provisioned for this network. [default]
Control.LUR.FailedRegistration.ShortCode 1000 [default]
Control.LUR.NormalRegistration.Message (disabled) [default]
Control.LUR.NormalRegistration.ShortCode 0000 [default]
Control.LUR.OpenRegistration .*
Control.LUR.OpenRegistration.Message Welcome to the test network.
Your IMSI is [default]
Control.LUR.OpenRegistration.Reject (disabled) [default]
Control.LUR.OpenRegistration.ShortCode 101 [default]
Control.LUR.QueryClassmark 0 [default]
Control.LUR.QueryIMEI 0 [default]
Control.LUR.RegistrationMessageFrequency FIRST [default]
Control.LUR.SendTMSIs 0 [default]
Control.LUR.UnprovisionedRejectCause 0x04 [default]
Control.Reporting.PhysStatusTable /var/run/ChannelTable.db
[default]
Control.Reporting.StatsTable /var/log/OpenBTSStats.db [default]
Control.Reporting.TMSITable /var/run/TMSITable.db [default]
Control.Reporting.TransactionTable /var/run/TransactionTable.db
[default]
Control.SMSCB.Table (disabled) [default]
Control.TMSITable.MaxAge 576 [default]
Control.VEA 0 [default]
```

GGSN.DNS (disabled) [default]
GGSN.Firewall.Enable 1 [default]
GGSN.IP.TossDuplicatePackets 0 [default]
GGSN.MS.IP.Base 192.168.99.1 [default]
GGSN.MS.IP.MaxCount 254 [default]
GGSN.MS.IP.Route (disabled) [default]
GGSN.ShellScript (disabled) [default]
GPRS.CellOptions.T3168Code 5 [default]
GPRS.CellOptions.T3192Code 0 [default]
GPRS.ChannelCodingControl.RSSI -40 [default]
GPRS.Channels.Min.C0 2 [default]
GPRS.Channels.Min.CN 0 [default]
GPRS.Enable 0 [default]
GPRS.LocalTLLI.Enable 1 [default]
GPRS.Multislot.Max.Downlink 3 [default]
GPRS.Multislot.Max.Uplink 2 [default]
GPRS.NMO 2 [default]
GPRS.RAC 0 [default]
GPRS.RA_COLOUR 0 [default]
GPRS.Reassign.Enable 1 [default]
GPRS.TBF.EST 1 [default]
GPRS.TBF.Retry 1 [default]
GSM.BTS.RADIO_LINK_TIMEOUT 15 [default]
GSM.CallerID.Source auto [default]
GSM.CellOptions.RADIO-LINK-TIMEOUT 15 [default]
GSM.CellSelection.CELL-RESELECT-HYSTERESIS 3 [default]
GSM.CellSelection.NCCsPermitted 0 [default]
GSM.CellSelection.NECI 1 [default]
GSM.Channels.ClsFirst 0 [default]
GSM.Channels.NumCls auto [default]
GSM.Channels.NumC7s auto [default]
GSM.Channels.SDCCHReserve 0 [default]
GSM.Cipher.CCHBER 0 [default]
GSM.Cipher.Encrypt 0 [default]
GSM.Cipher.RandomNeighbor 0 [default]
GSM.Cipher.ScrambleFiller 0 [default]

GSM.Handover.FailureHoldoff 20 [default]
GSM.Handover.Margin 15 [default]
GSM.Handover.Ny1 50 [default]
GSM.Identity.BSIC.BCC 2 [default]
GSM.Identity.BSIC.NCC 0 [default]
GSM.Identity.CI 10 [default]
GSM.Identity.LAC 1000 [default]
GSM.Identity.MCC 602
GSM.Identity.MNC 07
GSM.Identity.ShortName SkyComm
GSM.MS.Power.Damping 75 [default]
GSM.MS.Power.Max 33 [default]
GSM.MS.Power.Min 5 [default]
GSM.MS.TA.Damping 50 [default]
GSM.MS.TA.Max 62 [default]
GSM.MaxSpeechLatency 2 [default]
GSM.Neighbors (disabled) [default]
GSM.Neighbors.NumToSend 31 [default]
GSM.RACH.AC 0x0400 [default]
GSM.RACH.MaxRetrans 1 [default]
GSM.RACH.TxInteger 14 [default]
GSM.Radio.ARFCNs 1 [default]
GSM.Radio.Band 900 [default]
GSM.Radio.C0 51 [default]
GSM.Radio.MaxExpectedDelaySpread 4 [default]
GSM.Radio.PowerManager.MaxAttendB 0
GSM.Radio.PowerManager.MinAttendB 0 [default]
GSM.Radio.RSSITarget -50 [default]
GSM.Radio.SNRTarget 10 [default]
GSM.ShowCountry 0 [default]
GSM.SpeechBuffer 1 [default]
GSM.Timer.Handover.Holdoff 10 [default]
GSM.Timer.T3109 30000 [default]
GSM.Timer.T3212 0 [default]
Log.Alarms.Max 20 [default]
Log.Level NOTICE [default]

Peering.Neighbor.RefreshAge 60 [default]
Peering.NeighborTable.Path /var/run/NeighborTable.db [default]
Peering.Port 16001 [default]
RTP.Range 98 [default]
RTP.Start 16484 [default]
SIP.DTMF.RFC2833 1 [default]
SIP.DTMF.RFC2833.PayloadType 101 [default]
SIP.DTMF.RFC2976 0 [default]
SIP.Local.IP 127.0.0.1 [default]
SIP.Local.Port 5062 [default]
SIP.Proxy.Registration 127.0.0.1:5064 [default]
SIP.Proxy.SMS 127.0.0.1:5063 [default]
SIP.Proxy.Speech 127.0.0.1:5060 [default]
SIP.Proxy.USSD (disabled) [default]
SIP.RFC3428.NoTrying 0 [default]
SIP.SMSC smsc [default]
SMS.FakeSrcSMSC 0000 [default]
SMS.MIMEType application/vnd.3gpp.sms [default]
TRX.IP 127.0.0.1 [default]

Appendix C: Erlang B Table-Blocked Calls Cleared Model

Table B-1 shows the amount of traffic (in Erlang) carried by a given channels n for different blocking probability values.

Table C-1: Erlang B Table

n	P_B (Blocking Probability)												
	0.01%	0.02%	0.03%	0.05%	0.1%	0.2%	0.3%	0.4%	0.5%	0.6%	0.7%	0.8%	0.9%
1	0.0001	0.0002	0.0003	0.0005	0.0010	0.0020	0.0030	0.0040	0.0050	0.0060	0.0070	0.0081	0.0091
2	0.0142	0.0202	0.0248	0.0321	0.0458	0.0653	0.0806	0.0937	0.105	0.116	0.126	0.135	0.1443
3	0.0868	0.110	0.127	0.152	0.194	0.249	0.289	0.321	0.349	0.374	0.397	0.418	0.4374
4	0.235	0.282	0.315	0.362	0.439	0.535	0.602	0.656	0.701	0.741	0.777	0.810	0.8415
5	0.452	0.527	0.577	0.649	0.762	0.900	0.994	1.07	1.13	1.19	1.24	1.28	1.326
6	0.728	0.832	0.900	0.996	1.15	1.33	1.45	1.54	1.62	1.69	1.75	1.81	1.867
7	1.05	1.19	1.27	1.39	1.58	1.80	1.95	2.06	2.16	2.24	2.31	2.38	2.448
8	1.42	1.58	1.69	1.83	2.05	2.31	2.48	2.62	2.73	2.83	2.91	2.99	3.069
9	1.83	2.01	2.13	2.30	2.56	2.85	3.05	3.21	3.33	3.44	3.54	3.63	3.7110
10	2.26	2.47	2.61	2.80	3.09	3.43	3.65	3.82	3.96	4.08	4.19	4.29	4.3811
11	2.72	2.96	3.12	3.33	3.65	4.02	4.27	4.45	4.61	4.74	4.86	4.97	5.07
12	3.21	3.47	3.65	3.88	4.23	4.64	4.90	5.11	5.28	5.43	5.55	5.67	5.78
13	3.71	4.01	4.19	4.45	4.83	5.27	5.56	5.78	5.96	6.12	6.26	6.39	6.50
14	4.24	4.56	4.76	5.03	5.45	5.92	6.23	6.47	6.66	6.83	6.98	7.12	7.24
15	4.78	5.12	5.34	5.63	6.08	6.58	6.91	7.17	7.38	7.56	7.71	7.86	7.99
16	5.34	5.70	5.94	6.25	6.72	7.26	7.61	7.88	8.10	8.29	8.46	8.61	8.75
17	5.91	6.30	6.55	6.88	7.38	7.95	8.32	8.60	8.83	9.03	9.21	9.37	9.52
18	6.50	6.91	7.17	7.52	8.05	8.64	9.03	9.33	9.58	9.79	9.98	10.1	10.3
19	7.09	7.53	7.80	8.17	8.72	9.35	9.76	10.1	10.3	10.6	10.7	10.9	11.1
20	7.70	8.16	8.44	8.83	9.41	10.1	10.5	10.8	11.1	11.3	11.5	11.7	11.9
21	8.32	8.79	9.10	9.50	10.1	10.8	11.2	11.6	11.9	12.1	12.3	12.5	12.7
22	8.95	9.44	9.76	10.2	10.8	11.5	12.0	12.3	12.6	12.9	13.1	13.3	13.5
23	9.58	10.1	10.4	10.9	11.5	12.3	12.7	13.1	13.4	13.7	13.9	14.1	14.3
24	10.2	10.8	11.1	11.6	12.2	13.0	13.5	13.9	14.2	14.5	14.7	14.9	15.1
25	10.9	11.4	11.8	12.3	13.0	13.8	14.3	14.7	15.0	15.3	15.5	15.7	15.9
26	11.5	12.1	12.5	13.0	13.7	14.5	15.1	15.5	15.8	16.1	16.3	16.6	16.8
27	12.2	12.8	13.2	13.7	14.4	15.3	15.8	16.3	16.6	16.9	17.2	17.4	17.6
28	12.9	13.5	13.9	14.4	15.2	16.1	16.6	17.1	17.4	17.7	18.0	18.2	18.4
29	13.6	14.2	14.6	15.1	15.9	16.8	17.4	17.9	18.2	18.5	18.8	19.1	19.3
30	14.2	14.9	15.3	15.9	16.7	17.6	18.2	18.7	19.0	19.4	19.6	19.9	20.1
31	14.9	15.6	16.0	16.6	17.4	18.4	19.0	19.5	19.9	20.2	20.5	20.7	21.0
32	15.6	16.3	16.8	17.3	18.2	19.2	19.8	20.3	20.7	21.0	21.3	21.6	21.8
33	16.3	17.0	17.5	18.1	19.0	20.0	20.6	21.1	21.5	21.9	22.2	22.4	22.7
34	17.0	17.8	18.2	18.8	19.7	20.8	21.4	21.9	22.3	22.7	23.0	23.3	23.5
35	17.8	18.5	19.0	19.6	20.5	21.6	22.2	22.7	23.2	23.5	23.8	24.1	24.4
36	18.5	19.2	19.7	20.3	21.3	22.4	23.1	23.6	24.0	24.4	24.7	25.0	25.3
37	19.2	20.0	20.5	21.1	22.1	23.2	23.9	24.4	24.8	25.2	25.6	25.9	26.1
38	19.9	20.7	21.2	21.9	22.9	24.0	24.7	25.2	25.7	26.1	26.4	26.7	27.0
39	20.6	21.5	22.0	22.6	23.7	24.8	25.5	26.1	26.5	26.9	27.3	27.6	27.9
40	21.4	22.2	22.7	23.4	24.4	25.6	26.3	26.9	27.4	27.8	28.1	28.5	28.7
41	22.1	23.0	23.5	24.2	25.2	26.4	27.2	27.8	28.2	28.6	29.0	29.3	29.6
42	22.8	23.7	24.2	25.0	26.0	27.2	28.0	28.6	29.1	29.5	29.9	30.2	30.5
43	23.6	24.5	25.0	25.7	26.8	28.1	28.8	29.4	29.9	30.4	30.7	31.1	31.4
44	24.3	25.2	25.8	26.5	27.6	28.9	29.7	30.3	30.8	31.2	31.6	31.9	32.3
45	25.1	26.0	26.6	27.3	28.4	29.7	30.5	31.1	31.7	32.1	32.5	32.8	33.1
46	25.8	26.8	27.3	28.1	29.3	30.5	31.4	32.0	32.5	33.0	33.4	33.7	34.0
47	26.6	27.5	28.1	28.9	30.1	31.4	32.2	32.9	33.4	33.8	34.2	34.6	34.9
48	27.3	28.3	28.9	29.7	30.9	32.2	33.1	33.7	34.2	34.7	35.1	35.5	35.8
49	28.1	29.1	29.7	30.5	31.7	33.0	33.9	34.6	35.1	35.6	36.0	36.4	36.7
50	28.9	29.9	30.5	31.3	32.5	33.9	34.8	35.4	36.0	36.5	36.9	37.2	37.6
	0.01%	0.02%	0.03%	0.05%	0.1%	0.2%	0.3%	0.4%	0.5%	0.6%	0.7%	0.8%	0.9%
N	B												

n	P _B												
	1.0%	1.2%	1.5%	2%	3%	5%	7%	10%	15%	20%	30%	40%	50%
1	0.0101	0.0121	0.0152	0.0204	0.0309	0.0526	0.753	0.111	0.176	0.250	0.429	0.667	1.00
2	0.153	0.168	0.190	0.223	0.282	0.381	0.470	0.595	0.796	1.00	1.45	2.00	2.73
3	0.455	0.489	0.535	0.602	0.715	0.899	1.06	1.27	1.60	1.93	2.63	3.48	4.59
4	0.869	0.922	0.992	1.09	1.26	1.52	1.75	2.05	2.50	2.95	3.89	5.02	6.50
5	1.36	1.43	1.52	1.66	1.88	2.22	2.50	2.88	3.45	4.01	5.19	6.60	8.44
6	1.91	2.00	2.11	2.28	2.54	2.96	3.30	3.76	4.44	5.11	6.51	8.19	10.4
7	2.50	2.60	2.74	2.94	3.25	3.74	4.14	4.67	5.46	6.23	7.86	9.80	12.4
8	3.13	3.25	3.40	3.63	3.99	4.54	5.00	5.60	6.50	7.37	9.21	11.4	14.3
9	3.78	3.92	4.09	4.34	4.75	5.37	5.88	6.55	7.55	8.52	10.6	13.0	16.3
10	4.46	4.61	4.81	5.08	5.53	6.22	6.78	7.51	8.62	9.68	12.0	14.7	18.3
11	5.16	5.32	5.54	5.84	6.33	7.08	7.69	8.49	9.69	10.9	13.3	16.3	20.3
12	5.88	6.05	6.29	6.61	7.14	7.95	8.61	9.47	10.8	12.0	14.7	18.0	22.2
13	6.61	6.80	7.05	7.40	7.97	8.83	9.54	10.5	11.9	13.2	16.1	19.6	24.2
14	7.35	7.56	7.82	8.20	8.80	9.73	10.5	11.5	13.0	14.4	17.5	21.2	26.2
15	8.11	8.33	8.61	9.01	9.65	10.6	11.4	12.5	14.1	15.6	18.9	22.9	28.2
16	8.88	9.11	9.41	9.83	10.5	11.5	12.4	13.5	15.2	16.8	20.3	24.5	30.2
17	9.65	9.89	10.2	10.7	11.4	12.5	13.4	14.5	16.3	18.0	21.7	26.2	32.2
18	10.4	10.7	11.0	11.5	12.2	13.4	14.3	15.5	17.4	19.2	23.1	27.8	34.2
19	11.2	11.5	11.8	12.3	13.1	14.3	15.3	16.6	18.5	20.4	24.5	29.5	36.2
20	12.0	12.3	12.7	13.2	14.0	15.2	16.3	17.6	19.6	21.6	25.9	31.2	38.2
21	12.8	13.1	13.5	14.0	14.9	16.2	17.3	18.7	20.8	22.8	27.3	32.8	40.2
22	13.7	14.0	14.3	14.9	15.8	17.1	18.2	19.7	21.9	24.1	28.7	34.5	42.1
23	14.5	14.8	15.2	15.8	16.7	18.1	19.2	20.7	23.0	25.3	30.1	36.1	44.1
24	15.3	15.6	16.0	16.6	17.6	19.0	20.2	21.8	24.2	26.5	31.6	37.8	46.1
25	16.1	16.5	16.9	17.5	18.5	20.0	21.2	22.8	25.3	27.7	33.0	39.4	48.1
26	17.0	17.3	17.8	18.4	19.4	20.9	22.2	23.9	26.4	28.9	34.4	41.1	50.1
27	17.8	18.2	18.6	19.3	20.3	21.9	23.2	24.9	27.6	30.2	35.8	42.8	52.1
28	18.6	19.0	19.5	20.2	21.2	22.9	24.2	26.0	28.7	31.4	37.2	44.4	54.1
29	19.5	19.9	20.4	21.0	22.1	23.8	25.2	27.1	29.9	32.6	38.6	46.1	56.1
30	20.3	20.7	21.2	21.9	23.1	24.8	26.2	28.1	31.0	33.8	40.0	47.7	58.1
31	21.2	21.6	22.1	22.8	24.0	25.8	27.2	29.2	32.1	35.1	41.5	49.4	60.1
32	22.0	22.5	23.0	23.7	24.9	26.7	28.2	30.2	33.3	36.3	42.9	51.1	62.1
33	22.9	23.3	23.9	24.6	25.8	27.7	29.3	31.3	34.4	37.5	44.3	52.7	64.1
34	23.8	24.2	24.8	25.5	26.8	28.7	30.3	32.4	35.6	38.8	45.7	54.4	66.1
35	24.6	25.1	25.6	26.4	27.7	29.7	31.3	33.4	36.7	40.0	47.1	56.0	68.1
36	25.5	26.0	26.5	27.3	28.6	30.7	32.3	34.5	37.9	41.2	48.6	57.7	70.1
37	26.4	26.8	27.4	28.3	29.6	31.6	33.3	35.6	39.0	42.4	50.0	59.4	72.1
38	27.3	27.7	28.3	29.2	30.5	32.6	34.4	36.6	40.2	43.7	51.4	61.0	74.1
39	28.1	28.6	29.2	30.1	31.5	33.6	35.4	37.7	41.3	44.9	52.8	62.7	76.1
40	29.0	29.5	30.1	31.0	32.4	34.6	36.4	38.8	42.5	46.1	54.2	64.4	78.1
41	29.9	30.4	31.0	31.9	33.4	35.6	37.4	39.9	43.6	47.4	55.7	66.0	80.1
42	30.8	31.3	31.9	32.8	34.3	36.6	38.4	40.9	44.8	48.6	57.1	67.7	82.1
43	31.7	32.2	32.8	33.8	35.3	37.6	39.5	42.0	45.9	49.9	58.5	69.3	84.1
44	32.5	33.1	33.7	34.7	36.2	38.6	40.5	43.1	47.1	51.1	59.9	71.0	86.1
45	33.4	34.0	34.6	35.6	37.2	39.6	41.5	44.2	48.2	52.3	61.3	72.7	88.1
46	34.3	34.9	35.6	36.5	38.1	40.5	42.6	45.2	49.4	53.6	62.8	74.3	90.1
47	35.2	35.8	36.5	37.5	39.1	41.5	43.6	46.3	50.6	54.8	64.2	76.0	92.1
48	36.1	36.7	37.4	38.4	40.0	42.5	44.6	47.4	51.7	56.0	65.6	77.7	94.1
49	37.0	37.6	38.3	39.3	41.0	43.5	45.7	48.5	52.9	57.3	67.0	79.3	96.1
50	37.9	38.5	39.2	40.3	41.9	44.5	46.7	49.6	54.0	58.5	68.5	81.0	98.1
N	1.0%	1.2%	1.5%	2%	3%	5%	7%	10%	15%	20%	30%	40%	50%

n	P _B												
	0.01%	0.02%	0.03%	0.05%	0.1%	0.2%	0.3%	0.4%	0.5%	0.6%	0.7%	0.8%	0.9%
50	28.9	29.9	30.5	31.3	32.5	33.9	34.8	35.4	36.0	36.5	36.9	37.2	37.6
51	29.6	30.6	31.3	32.1	33.3	34.7	35.6	36.3	36.9	37.3	37.8	38.1	38.5
52	30.4	31.4	32.0	32.9	34.2	35.6	36.5	37.2	37.7	38.2	38.6	29.0	39.4
53	31.2	32.2	32.8	33.7	35.0	36.4	37.3	38.0	38.6	39.1	39.5	39.9	40.3
54	31.9	33.0	33.6	34.5	35.8	37.2	38.2	38.9	39.5	40.0	40.4	40.8	41.2
55	32.7	33.8	34.4	35.3	36.6	38.1	39.0	39.8	40.4	40.9	41.3	41.7	42.1
56	33.5	34.6	35.2	36.1	37.5	38.9	39.9	40.6	41.2	41.7	42.2	42.6	43.0
57	34.3	35.4	36.0	36.9	38.3	39.8	40.8	41.5	42.1	42.6	43.1	43.5	43.9
58	35.1	36.2	36.8	37.8	39.1	40.6	41.6	42.4	43.0	43.5	44.0	44.4	44.8
59	35.8	37.0	37.6	38.6	40.0	41.5	42.5	43.3	43.9	44.4	44.9	45.3	45.7
60	36.6	37.8	38.5	39.4	40.8	42.4	43.4	44.1	44.8	45.3	45.8	46.2	46.6
61	37.4	38.6	39.3	40.2	41.6	43.2	44.2	45.0	45.6	46.2	46.7	47.1	47.5
62	38.2	39.4	40.1	41.0	42.5	44.1	45.1	45.9	46.5	47.1	47.6	48.0	48.4
63	39.0	40.2	40.9	41.9	43.3	44.9	46.0	46.8	47.4	48.0	48.5	48.9	49.3
64	39.8	41.0	41.7	42.7	44.2	45.8	46.8	47.6	48.3	48.9	49.4	49.8	50.2
65	40.6	41.8	42.5	43.5	45.0	46.6	47.7	48.5	49.2	49.8	50.3	50.7	51.1
66	41.4	42.6	43.3	44.4	45.8	47.5	48.6	49.4	50.1	50.7	51.2	51.6	52.0
67	42.2	43.4	44.2	45.2	46.7	48.4	49.5	50.3	51.0	51.6	52.1	52.5	53.0
68	43.0	44.2	45.0	46.0	47.5	49.2	50.3	51.2	51.9	52.5	53.0	53.4	53.9
69	43.8	45.0	45.8	46.8	48.4	50.1	51.2	52.1	52.8	53.4	53.9	54.4	54.8
70	44.6	45.8	46.6	47.7	49.2	51.0	52.1	53.0	53.7	54.3	54.8	55.3	55.7
71	45.4	46.7	47.5	48.5	50.1	51.8	53.0	53.8	54.6	55.2	55.7	56.2	56.6
72	46.2	47.5	48.3	49.4	50.9	52.7	53.9	54.7	55.5	56.1	56.6	57.1	57.5
73	47.0	48.3	49.1	50.2	51.8	53.6	54.7	55.6	56.4	57.0	57.5	58.0	58.5
74	47.8	49.1	49.9	51.0	52.7	54.5	55.6	56.5	57.3	57.9	58.4	58.9	59.4
75	48.6	49.9	50.8	51.9	53.5	55.3	56.5	57.4	58.2	58.8	59.3	59.8	60.3
76	49.4	50.8	51.6	52.7	54.4	56.2	57.4	58.3	59.1	59.7	60.3	60.8	61.2
77	50.2	51.6	52.4	53.6	55.2	57.1	58.3	59.2	60.0	60.6	61.2	61.7	62.1
78	51.1	52.4	53.3	54.4	56.1	58.0	59.2	60.1	60.9	61.5	62.1	62.6	63.1
79	51.9	53.2	54.1	55.3	56.9	58.8	60.1	61.0	61.8	62.4	63.0	63.5	64.0
80	52.7	54.1	54.9	56.1	57.8	59.7	61.0	61.9	62.7	63.3	63.9	64.4	64.9
81	53.5	54.9	55.8	56.9	58.7	60.6	61.8	62.8	63.6	64.2	64.8	65.4	65.8
82	54.3	55.7	56.6	57.8	59.5	61.5	62.7	63.7	64.5	65.2	65.7	66.3	66.8
83	55.1	56.6	57.5	58.6	60.4	62.4	63.6	64.6	65.4	66.1	66.7	67.2	67.7
84	56.0	57.4	58.3	59.5	61.3	63.2	64.5	65.5	66.3	67.0	67.6	68.1	68.6
85	56.8	58.2	59.1	60.4	62.1	64.1	65.4	66.4	67.2	67.9	68.5	69.1	69.6
86	57.6	59.1	60.0	61.2	63.0	65.0	66.3	67.3	68.1	68.8	69.4	70.0	70.5
87	58.4	59.9	60.8	62.1	63.9	65.9	67.2	68.2	69.0	69.7	70.3	70.9	71.4
88	59.3	60.8	61.7	62.9	64.7	66.8	68.1	69.1	69.9	70.6	71.3	71.8	72.3
89	60.1	61.6	62.5	63.8	65.6	67.7	69.0	70.0	70.8	71.6	72.2	72.8	73.3
90	60.9	62.4	63.4	64.6	66.5	68.6	69.9	70.9	71.8	72.5	73.1	73.7	74.2
91	61.8	63.3	64.2	65.5	67.4	69.4	70.8	71.8	72.7	73.4	74.0	74.6	75.1
92	62.6	64.1	65.1	66.3	68.2	70.3	71.7	72.7	73.6	74.3	75.0	75.5	76.1
93	63.4	65.0	65.9	67.2	69.1	71.2	72.6	73.6	74.5	75.2	75.9	76.5	77.0
94	64.2	65.8	66.8	68.1	70.0	72.1	73.5	74.5	75.4	76.2	76.8	77.4	77.9
95	65.1	66.6	67.6	68.9	70.9	73.0	74.4	75.5	76.3	77.1	77.7	78.3	78.9
96	65.9	67.5	68.5	69.8	71.7	73.9	75.3	76.4	77.2	78.0	78.7	79.3	79.8
97	66.8	68.3	69.3	70.7	72.6	74.8	76.2	77.3	78.2	78.9	79.6	80.2	80.7
98	67.6	69.2	70.2	71.5	73.5	75.7	77.1	78.2	79.1	79.8	80.5	81.1	81.7
99	68.4	70.0	71.0	72.4	74.4	76.6	78.0	79.1	80.0	80.8	81.4	82.0	82.6
100	69.3	70.9	71.9	73.2	75.2	77.5	78.9	80.0	80.9	81.7	82.4	83.0	83.5
N	0.01%	0.02%	0.03%	0.05%	0.1%	0.2%	0.3%	0.4%	0.5%	0.6%	0.7%	0.8%	0.9%
	B												

n	P _B												
	1.0%	1.2%	1.5%	2%	3%	5%	7%	10%	15%	20%	30%	40%	50%
50	37.9	38.5	39.2	40.3	41.9	44.5	46.7	49.6	54.0	58.5	68.5	81.0	98.1
51	38.8	39.4	40.1	41.2	42.9	45.5	47.7	50.6	55.2	59.7	69.9	82.7	100.1
52	39.7	40.3	41.0	42.1	43.9	46.5	48.8	51.7	56.3	61.0	71.3	84.3	102.1
53	40.6	41.2	42.0	43.1	44.8	47.5	49.8	52.8	57.5	62.2	72.7	86.0	104.1
54	41.5	42.1	42.9	44.0	45.8	48.5	50.8	53.9	58.7	63.5	74.2	87.6	106.1
55	42.4	43.0	43.8	44.9	46.7	49.5	51.9	55.0	59.8	64.7	75.6	89.3	108.1
56	43.3	43.9	44.7	45.9	47.7	50.5	52.9	56.1	61.0	65.9	77.0	91.0	110.1
57	44.2	44.8	45.7	46.8	48.7	51.5	53.9	57.1	62.1	67.2	78.4	92.6	112.1
58	45.1	45.8	46.6	47.8	49.6	52.6	55.0	58.2	63.3	68.4	79.8	94.3	114.1
59	46.0	46.7	47.5	48.7	50.6	53.6	56.0	59.3	64.5	69.7	81.3	96.0	116.1
60	46.9	47.6	48.4	49.6	51.6	54.6	57.1	60.4	65.6	70.9	82.7	97.6	118.1
61	47.9	48.5	49.4	50.6	52.5	55.6	58.1	61.5	66.8	72.1	84.1	99.3	120.1
62	48.8	49.4	50.3	51.5	53.5	56.6	59.1	62.6	68.0	73.4	85.5	101.0	122.1
63	49.7	50.4	51.2	52.5	54.5	57.6	60.2	63.7	69.1	74.6	87.0	102.6	124.1
64	50.6	51.3	52.2	53.4	55.4	58.6	61.2	64.8	70.3	75.9	88.4	104.3	126.1
65	51.5	52.2	53.1	54.4	56.4	59.6	62.3	65.8	71.4	77.1	89.8	106.0	128.1
66	52.4	53.1	54.0	55.3	57.4	60.6	63.3	66.9	72.6	78.3	91.2	107.6	130.1
67	53.4	54.1	55.0	56.3	58.4	61.6	64.4	68.0	73.8	79.6	92.7	109.3	132.1
68	54.3	55.0	55.9	57.2	59.3	62.6	65.4	69.1	74.9	80.8	94.1	111.0	134.1
69	55.2	55.9	56.9	58.2	60.3	63.7	66.4	70.2	76.1	82.1	95.5	112.6	136.1
70	56.1	56.8	57.8	59.1	61.3	64.7	67.5	71.3	77.3	83.3	96.9	114.3	138.1
71	57.0	57.8	58.7	60.1	62.3	65.7	68.5	72.4	78.4	84.6	98.4	115.9	140.1
72	58.0	58.7	59.7	61.0	63.2	66.7	69.6	73.5	79.6	85.8	99.8	117.6	142.1
73	58.9	59.6	60.6	62.0	64.2	67.7	70.6	74.6	80.8	87.0	101.2	119.3	144.1
74	59.8	60.6	61.6	62.9	65.2	68.7	71.7	75.6	81.9	88.3	102.7	120.9	146.1
75	60.7	61.5	62.5	63.9	66.2	69.7	72.7	76.7	83.1	89.5	104.1	122.6	148.0
76	61.7	62.4	63.4	64.9	67.2	70.8	73.8	77.8	84.2	90.8	105.5	124.3	150.0
77	62.6	63.4	64.4	65.8	68.1	71.8	74.8	78.9	85.4	92.0	106.9	125.9	152.0
78	63.5	64.3	65.3	66.8	69.1	72.8	75.9	80.0	86.6	93.3	108.4	127.6	154.0
79	64.4	65.2	66.3	67.7	70.1	73.8	76.9	81.1	87.7	94.5	109.8	129.3	156.0
80	65.4	66.2	67.2	68.7	71.1	74.8	78.0	82.2	88.9	95.7	111.2	130.9	158.0
81	66.3	67.1	68.2	69.6	72.1	75.8	79.0	83.3	90.1	97.0	112.6	132.6	160.0
82	67.2	68.0	69.1	70.6	73.0	76.9	80.1	84.4	91.2	98.2	114.1	134.3	162.0
83	68.2	69.0	70.1	71.6	74.0	77.9	81.1	85.5	92.4	99.5	115.5	135.9	164.0
84	69.1	69.9	71.0	72.5	75.0	78.9	82.2	86.6	93.6	100.7	116.9	137.6	166.0
85	70.0	70.9	71.9	73.5	76.0	79.9	83.2	87.7	94.7	102.0	118.3	139.3	168.0
86	70.9	71.8	72.9	74.5	77.0	80.9	84.3	88.8	95.9	103.2	119.8	140.9	170.0
87	71.9	72.7	73.8	75.4	78.0	82.0	85.3	89.9	97.1	104.5	121.2	142.6	172.0
88	72.8	73.7	74.8	76.4	78.9	83.0	86.4	91.0	98.2	105.7	122.6	144.3	174.0
89	73.7	74.6	75.7	77.3	79.9	84.0	87.4	92.1	99.4	106.9	124.0	145.9	176.0
90	74.7	75.6	76.7	78.3	80.9	85.0	88.5	93.1	100.6	108.2	125.5	147.6	178.0
91	75.6	76.5	77.6	79.3	81.9	86.0	89.5	94.2	101.7	109.4	126.9	149.3	180.0
92	76.6	77.4	78.6	80.2	82.9	87.1	90.6	95.3	102.9	110.7	128.3	150.9	182.0
93	77.5	78.4	79.6	81.2	83.9	88.1	91.6	96.4	104.1	111.9	129.7	152.6	184.0
94	78.4	79.3	80.5	82.2	84.9	89.1	92.7	97.5	105.3	113.2	131.2	154.3	186.0
95	79.4	80.3	81.5	83.1	85.8	90.1	93.7	98.6	106.4	114.4	132.6	155.9	188.0
96	80.3	81.2	82.4	84.1	86.8	91.1	94.8	99.7	107.6	115.7	134.0	157.6	190.0
97	81.2	82.2	83.4	85.1	87.8	92.2	95.8	100.8	108.8	116.9	135.5	159.3	192.0
98	82.2	83.1	84.3	86.0	88.8	93.2	96.9	101.9	109.9	118.2	136.9	160.9	194.0
99	83.1	84.1	85.3	87.0	89.8	94.2	97.9	103.0	111.1	119.4	138.3	162.6	196.0
100	84.1	85.0	86.2	88.0	90.8	95.2	99.0	104.1	112.3	120.6	139.7	164.3	198.0
N	1.0%	1.2%	1.5%	2%	3%	5%	7%	10%	15%	20%	30%	40%	50%
	B												