BLUETOOTH/WLAN RECEIVER DESIGN METHODOLOGY AND IC IMPLEMENTATIONS

A Dissertation

by

AHMED AHMED ELADAWY EMIRA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2003

Major Subject: Electrical Engineering

BLUETOOTH/WLAN RECEIVER DESIGN METHODOLOGY AND IC IMPLEMENTATIONS

A Dissertation

by

AHMED AHMED ELADAWY EMIRA

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Approved as to style and content by:

Edgar Sánchez-Sinencio (Chair of Committee)	José Silva-Martínez (Member)
Aydin Karsilayan (Member)	Scott Miller (Member)
Duncan Walker (Member)	Chanan Singh (Head of Department)

December 2003

Major Subject: Electrical Engineering

ABSTRACT

Bluetooth/WLAN Receiver Design Methodology and

IC Implementations. (December 2003)

Ahmed Ahmed Eladawy Emira, B.Eng., Cairo University, Egypt;

M.Eng., Cairo University, Egypt

Chair of Advisory Committee: Dr. Edgar Sánchez-Sinencio

Emerging technologies such as Bluetooth and 802.11b (Wi-Fi) have fuelled the growth of the short-range communication industry. Bluetooth, the leading WPAN (wireless personal area network) technology, was designed primarily for cable replacement applications. The first generation Bluetooth products are focused on providing low-cost radio connections among personal electronic devices. In the WLAN (wireless local area network) arena, Wi-Fi appears to be the superior product. Wi-Fi is designed for high speed internet access, with higher radio power and longer distances. Both technologies use the same 2.4GHz ISM band. The differences between Bluetooth and Wi-Fi standard features lead to a natural partitioning of applications. Nowadays, many electronics devices such as laptops and PDAs, support both Bluetooth and Wi-Fi standards to cover a wider range of applications. The cost of supporting both standards, however, is a major concern. Therefore, a dual-mode transceiver is essential to keep the size and cost of such system transceivers at a minimum.

A fully integrated low-IF Bluetooth receiver is designed and implemented in a low cost, main stream 0.35μm CMOS technology. The system includes the RF front end, frequency synthesizer and baseband blocks. It has -82dBm sensitivity and draws 65mA current. This project involved six Ph.D. students and I was in charge of the design of the channel selection complex filter.

In the Bluetooth transmitter, a frequency modulator with fine frequency steps is needed to generate the GFSK signal that has ±160kHz frequency deviation. A low power ROM-less direct digital frequency synthesizer (DDFS) is designed to implement the frequency modulation. The DDFS can be used for any frequency or phase modulation communication systems that require fast frequency switching with fine frequency steps.

Another contribution is the implementation of a dual-mode 802.11b/Bluetooth receiver in IBM 0.25μm BiCMOS process. Direct-conversion architecture was used for both standards to achieve maximum level of integration and block sharing. I was honored to lead the efforts of seven Ph.D. students in this project. I was responsible for system level design as well as the design of the variable gain amplifier. The receiver chip consumes 45.6/41.3mA and the sensitivity is -86/-91dBm.

DEDICATION

To my Parents,

To my Brother, and Sister,

And to my fiancée Eman

For their love and support

ACKNOWLEDGEMENTS

A journey is easier when you travel together. Interdependence is certainly more valuable than independence. This dissertation is the result of four years of work whereby I have been accompanied and supported by many people. It is a pleasant aspect to now have the opportunity to express my gratitude for all of them.

With a deep sense of gratitude, I wish to express my sincere thanks to my supervisor, Dr. Edgar Sánchez-Sinencio, for his immense help in planning and executing the work in a timely manner. The confidence and dynamism with which Dr. Sánchez guided the work requires no elaboration. His company and assurance at the time of crisis will always be remembered. His technical and editorial advice was essential to the completion of this dissertation and taught me innumerable lessons and insights on the workings of academic research in general.

I am also grateful to Dr. Sherif Embabi and his group at Texas Instruments for sharing much of their valuable time to review the design of the Bluetooth and Chameleon receiver projects. Their comments and suggestions were very useful in avoiding potential design problems and improving the performance of the receiver. I would also like to express my gratitude to Dr. José Silva-Martínez and Dr. Aydin Karsilayan who were always there to answer my technical questions throughout my entire Ph.D. program. Thanks also go to my committee members: Dr. Scott Miller and Dr. Duncan Walker for their time and valuable comments on my research.

I am fortunate to have had the opportunity to work with a group of energetic people in the Analog and Mixed Signal Center (AMSC). Special thanks and words of appreciation are due to my friend, research partner, and ex roommate Ahmed Nader

Mohieldin. I was his roommate during the first two years of my Ph.D. He made my arrival to College Station a lot smoother than I thought it would be. He was like my family away from home. On the technical side, Ahmed and I worked together on three fruitful projects, which constituted a major part of my Ph.D. I learned much from our numerous technical discussions. I also enjoyed the company of many students with whom I had very interesting technical discussions. I had the most wonderful learning experience working with a team of very active and bright Ph.D. students in the Bluetooth and the Chameleon projects, which included Alberto Valdes-Garcia, Ari Valero-Lopez, Wenjun Sheng, Bo Xia, Sung Tung Moon, Chunyu Xin, and Ahmed Nader Mohieldin. Their talent and determination was behind the success of the two projects. I am also grateful to my friend and colleague, Faisal Abdellatif Elseddeek, for good times we spent together as roommates. He provided me the emotional support I much needed at the last stage of my Ph.D.

Finally, I will never find words enough to express the gratitude that I owe to my late father, my mother, my sister Sahar, my brother Mohamed, and my fiancée Eman. Their tender love and support have always been the cementing force for building the blocks of my career. The all round support rendered by them provided the much needed stimulant to sail through the phases of stress and strain.

TABLE OF CONTENTS

Page

CHAPTER Page

CHAPTER Page

LIST OF FIGURES

Page

Page

Page

Page

Page

Page

Page

LIST OF TABLES

Page

Page

CHAPTER I

INTRODUCTION

Stimulated by the increased consumer and commercial users demand or wireless communications applications, the advancement of sophistication in wireless circuit design has progressed at an unprecedented speed in recent years. The last few years have witnessed a remarkable miniaturization of portable equipment, a similar extension of battery life, and almost an order of magnitude in retail prices. Two ubiquitous examples of this phenomenon are the wireless local area networks (WLANs) and wireless personal area networks (WPANs).

1.1. Background and Motivation

Although not well known among everyday wireless communication consumers, commercial wireless communications technology has also experienced a similar impetus in its advancement, both in miniaturization and cost. In portable wireless communication industry it is well known that the design of the RF transceiver is usually the key element that determines the cost, size, and useful battery life of the equipment, as well as how the equipment is used.

One of the major forces sustaining continual research and development in the wireless communication arena is the general user acceptance of wireless communications standards. These standards provide coalescence within the industry to invest in the creation of suitable chip sets for the manufacturing of consumer subscriber units. The resulting high volume drives down the cost of the final product, and industry is able to

---

This dissertation follows the style of *IEEE Journal of Solid-State Circuits*.

take advantage of the same trends that have dramatically lowered prices in the PC industry.

The main goal of this dissertation is to present system level and building block level solutions for two of the most widespread short-range communications standards, Bluetooth and 802.11b (Wi-Fi). The motivation is to bridge the gap between the circuit and wireless vision in wireless design. Although the dissertation focuses on Bluetooth and Wi-Fi radio implementations, many design aspects can be generalized to most other radio implementations. The Bluetooth and Wi-Fi standards operate at RF frequency 2.4GHz that is also used by microwave ovens and cordless phones. While Wi-Fi is meant to be wireless version of Ethernet, Bluetooth is designed to replace cables that are used for relatively low speed data transfer. Wi-Fi offers data rates up to 11Mbit/s while Bluetooth's raw data rate is 1Mbits/s. Bluetooth is supposed to be lower cost than Wi-Fi. In this dissertation, the design of a low-IF Bluetooth receiver fabricated in a low cost mainstream CMOS technology is presented. On the other hand, there are many devices that require having both standards implanted. In such cases, the integration of both standards into a single chip would be cheaper than having two separate chip solutions. For this purpose, a dual-mode Bluetooth/Wi-Fi chip solutions is also described.

1.2. Dissertation Overview

Chapter II presents short-range communications methods. Wired and wireless communications standards are introduced with some focus on wireless standards since it is the topic of the following chapters. In particular, details about Bluetooth and 802.11b (Wi-Fi) standards are given to get the reader acquainted with these standards before presenting the receivers design specifications and implementations in later chapters.

Chapter III presents Bluetooth possible receiver architectures and discusses the pros and cons of each architecture. The proposed low-IF Bluetooth architecture is discussed with some detail. As part of the Bluetooth receiver, detailed design, implementation, and measurement results of channel selection complex filter are presented. In the Bluetooth transmitter, on the other hand, direct digital frequency synthesizer (DDFS) is used as the frequency modulator. DDFS design and implementation is discussed in details in chapter IV. A dual-mode 802.11b/Bluetooth receiver design is presented in chapter V. The dual-mode receiver is named "Chameleon" for its ability to change operation modes. An overview of previous dual-mode receiver implementations is included. The proposed direct conversion architecture is presented. Effects of receiver non-idealities of its performance are simulated and quantified. Detailed design procedure starting from standard specifications to receiver specifications to building block requirements is presented. Chapter VI gives the design of the Chameleon receiver building blocks. Variable gain amplifier (VGA) design is presented in details. Measured characteristics of the receiver blocks are also included. In chapter VII, experimental results of the Chameleon receiver are presented. Test setup as well as measurement procedures are illustrated. Finally, chapter VIII summarizes the main contributions of this research work. It should be mentioned that the Bluetooth and Chameleon projects were carried-out by a team of seven Ph.D. students. This dissertation covers partially the entire design process and implementations, but the details of some system aspects and building blocks developed and proposed by the author are emphasized in the essence of this dissertation.

CHAPTER II

SHORT-RANGE COMMUNICATION STANDARDS

In recent years mobile digital devices such as personal digital assistants (PDAs), mobile phones, digital cameras, and laptops have penetrated the consumer market. All these devices require powerful short-range communication method for data exchange among each other, connections with printers or local area network (LAN) access. Basically, the communication methods can be based on cable connections, radio links, or infrared links as illustrated in Fig. 2.1. Since each has its individual strengths and weaknesses, each found its way into various products.

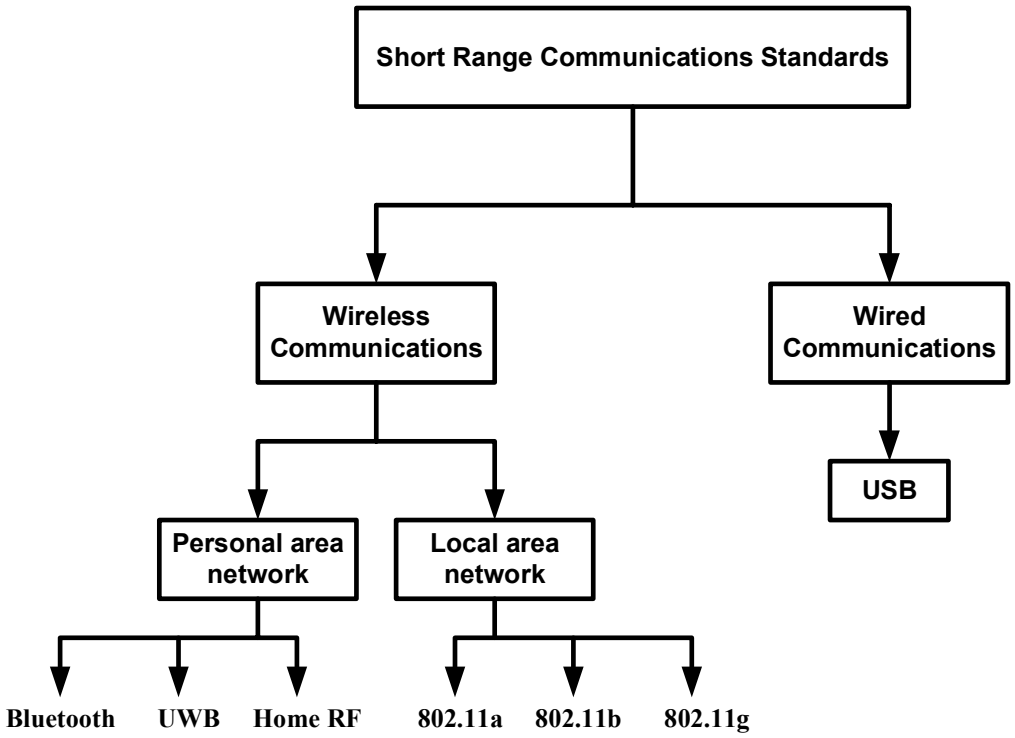


Fig. 2.1 Short range communication standards

2.1. Short-Range Wired Communications

Data exchange via cables is a well established method; universal serial bus (USB) has become widely used standard interface. USB excels due to its high baud rates up to 480Mb/s, but suffers from its limited mobility due to cable connection. Therefore, USB is best for applications that require stable high-performance connections for transmission of high data volumes, where mobility is not very important. An example of application would be the connection of a videoconferencing camera to your laptop.

2.2. Short-Range Wireless Communications

Contrary to USB, infrared transmission based on the Infrared Data Association (IrDA) standard enables fast connection establishment due to its point-and-shoot characteristic. Together with the high baud rates up to 16Mb/s, this makes transmission well suited to applications that require high performance as hoc point-to-point connections. Examples would include downloading of pictures from your digital camera to your laptop or paying for your meal in your company's cafeteria with your mobile phone via IrDA port. IrDA standards have been there for decades and are widely implemented in laptops, computers and PDAs. But until recently, either the cost was too high, or in the case of infrared, the technology was too difficult to use.

Radio-based short-range wireless (SRW) communication is an alternative class of emerging technologies designed primarily for indoor use over very short distances. It is intended to provide fast (tens or hundreds of megabits per second) and low cost, cable-free connections to the internet. SRW features transmission powers of several microwatts

up to milliwatts yielding a communication range between 10 and 100 meter. SRW will provide connectivity to portable devices such as laptops, PDAs, cell phones and others.

Short-range communications standards fall into two broad but overlapping categories: personal area networks (PAN) and local area networks (LAN).

Wireless PAN technologies emphasize low cost and low power consumption, usually at the expense of range and peak speed. In a typical wireless PAN application, a short wireless link, typically under 10 meters, replaces a computer serial cable or USB cable. Standards, such as Bluetooth and HomeRF, have been created to regulate short-range wireless communications. Bluetooth has appeared recently in many mobile devices. Bluetooth can transmit data through solid nonmetal objects and supports a nominal link range of 10cm-10m at a moderate baud rate up to 720kb/s (raw data rate is 1Mb/s) [1]. An optional high power mode in the current specifications allows for ranges up to 100m. Because of the nature of radio, Bluetooth is a point to multipoint communication system, which supports connections of two devices as well as ad hoc networking between several devices. But in order to prevent unauthorized access, Bluetooth requires sophisticated authentication and encryption mechanisms, which hamper fast connection establishment. Therefore, Bluetooth is best for applications that require stable point-to-point or point-to-multipoint connections for data exchange at moderate speeds, where mobility is a key requirement. Ultra-wideband (UWB) is an emerging new technology that shows great potential for SRW applications. Unlike conventional wireless communications systems that are carrier-based, UWB-based communication is baseband. It uses a series of short pulses that spread the energy of the signal from near DC to a few GHz.

Wireless LAN technologies, on the other hand, emphasize a higher peak speed and longer range at the expense of cost and power consumption. Typically, wireless LANs provide wireless links from portable laptops to a wired LAN access point. To date, 802.11b has gained acceptance rapidly as a wireless LAN standard. It has a nominal open-space range of 100m and a peak over-the-air speed of 11Mb/s. Users can expect maximum available speeds of about 5.5Mb/s. Other communication standards offer even higher data rates, like 802.11a and 802.11g. Table 2.1 compares between the leading radio-based short-range communication standards.

Table 2.1 Summary of characteristics of some leading WLAN/WPAN standards

Characteristic	Bluetooth	IEEE 802.11b	IEEE 802.11g	IEEE 802.11a	UWB
Standard version/status	V 1.1 (Low-Rate)	IEEE approved	Draft	IEEE approved	Draft
Maximum distance	10-100m	100m	100m	50m	10m
Frequency allocation	2.4GHz (ISM)	2.4GHz (ISM)	2.4GHz (ISM)	5GHz (UNII)	3.1-10.6GHz
Number of RF channels	79	3	3	12 (U.S.) 8 (EU) 4 (Japan)	1-15
Modulation	GFSK	QPSK (CCK)	OFDM	OFDM	BPSK, QPSK
Spreading	FH	DSSS CCK	OFDM	OFDM	(Multiband)
Maximum RF power	0-20dBm	30dBm (U.S.) 20dBm (EU) 10dBm (Japan)	30dBm (U.S.) 20dBm (EU) 10dBm (Japan)	17dBm, 24dBm, 30dBm	-41.3dBm/MHz
receiver sensitivity	-70dBm	-76dBm for 11Mb/s	-74dBm for 33Mb/s	-65dBm or 54Mb/s	-

2.3. What is Bluetooth?

Bluetooth technology was developed to create a short-range wireless voice and data link between a broad range of devices such as PCs, notebook computers, handhelds and PDAs (hereafter referred to as PDAs), Smart Phones, mobile phones and digital cameras. Consistent with its aim of operating in even the smallest battery-powered devices, the Bluetooth specification calls for a small form factor, low power consumption and low cost. The range and speed of the technology were kept intentionally low so as to ensure maximum battery life and minimum incremental cost for devices incorporating the technology. At its heart, Bluetooth is about creating a Wireless Personal Area Network (WPAN) consisting of all the Bluetooth-enabled electronic devices immediately surrounding a user, wherever that user may be located. This project is supported by a special interest group formed by hundreds of companies that lead technological development.

Specifications of Bluetooth provide the limitations and possibilities of the technology. Bluetooth operates in the 2.4GHz frequencies which is free for use to everyone globally. This frequency provides an effective data rate of 720 Kbit/s. There are multiple classes of transmission strengths which determine the use and the energy efficiency of these devices. The Bluetooth network protocol employs five layers which are implemented by software and hardware. They are:

(a) Application Programming Interface (API) – Allows the operating system (OS) and other applications to access the Bluetooth interface.

(b) Logical Link Control and Adaptation Protocol (L2CAP) – Maintains individual links to other devices in its transmission area.

(c) Link Manager – Responsible for establishing and terminating links and link security.

(d) Baseband – low level tasks including error correction.

(e) Physical – actual radio transmitter and antenna.

The topology for Bluetooth is very unique and this feature allows it to be completely scalable. The network link can be either point to point or can be point to multipoint, and devices participating in the network can be defined as master or slave. In the simplest network of one master device communicating with one slave device, a piconet is formed and all the bandwidth is dedicated to the link between the two devices. Larger networks can be formed with a single master and up to 7 slaves. The network architecture, however, allows up to 255 slaves to be in a standby mode. Bluetooth manages to achieve its functionality through its modulation and packeting scheme. This is also known as Gaussian Frequency Shift Keying (GFSK).

Bluetooth hops to one of 79 different channels (US and Europe), a repetitive process that keeps errors to a minimum. It is this packet hopping technique which leads to the interference with 802.11b technology. The channel hopping technology of Bluetooth is the secret behind its low power consumption, error correction and the distinct topology that it can support. Bluetooth divides the data to be sent into packets. Each packet is sent within a 625- microsecond slot. A frame is normally defined as a transmit and a receive slot, providing full duplex communication between a master and a slave in one time frame. To avoid noise and other interferences, Bluetooth hops to one of 79 different channels each time frame. The channel that it hops to is determined by the Master ID and the previous channel number. This algorithm is repeated again and again. For example if

there is severe noise between 2.408GHz and 2.410GHz, it will be avoided the majority of the time. There is little to no time contention between masters within reach of each other which might result in the masters picking up the same channels at the same time. Because of this channel hopping mechanism, interference is kept to a minimum despite extremely dense scatternets. Master devices can use this frame division to communicate with each slave on the piconet consecutively, with 1 frame each, or they can devote multiple frames to the same slave device, depending on the priority of the job at hand.

Bluetooth was originally conceived by Ericsson in 1994, when they began a study to examine alternatives to cables that linked mobile phone accessories. Ericsson already had a strong capability in short range wireless, having been a key pioneer of the European DECT cordless telecommunications standard, which had been largely based upon their earlier proprietary DCT900 technology. Out of their study was born the specification for Bluetooth wireless.

Bluetooth was named after Harald Blatand (or Bluetooth), a tenth century Danish Viking king who had united and controlled large parts of Scandinavia which are today Denmark and Norway. The name was chosen to highlight the potential of the technology to unify the telecommunications and computing industries - although it was chosen as an internal codename, and it was never at the time expected to survive as the name used in the commercial arena.

In February 1998, the Bluetooth SIG (Special Interest Group) was founded by a small core of major companies - IBM, Intel, Nokia, Toshiba and Ericsson - to work together to develop the technology and to subsequently promote its widespread commercial acceptance.

Six months later the core Promoter Members publicly announced the global SIG and invited other companies to join, with free access to the technology as Bluetooth adopters in return for commitment to support the Bluetooth specification. Adoption was rapid and 1998-1999 saw a boom in the market for Bluetooth conference organizers, and vast amounts of hype regarding the potential of the technology. In December 1999 it was announced that four more major companies had joined the SIG as Promoter Members, viz. Microsoft, Agere Systems (then Lucent), 3Com and Motorola.

The detailed Bluetooth specifications are available in [1]. Table 2.2 is a summary of the key radio specifications.

2.3.1. Bluetooth Operation

Bluetooth controls timing on the network by designating one of the devices as a master and the other as a slave. The master is simply the unit that initiates the communication link, and the other participants are slaves. When that link is later broken, the master/slave designations no longer apply. In fact, every Bluetooth device has both master and slave hardware. The network itself is termed a piconet, meaning small network. When there is only one slave, then the link is called point-to-point. A master can control up to seven active slaves in a point-to-multipoint configuration. Slaves communicate only with the master, never with each other directly. Timing is such that members of the piconet cannot transmit simultaneously, so these devices will not jam each other. Finally, communication across piconets can be realized if a Bluetooth device can be a slave in more than one piconet, or a master in one and a slave in another. Piconets configured in this manner are called scatternets. These various arrangements are depicted in Fig. 2.2 [2].

Table 2.2 Bluetooth radio specifications

Frequency Band		2400 – 2483.5 MHz
Duplex		Time Division
Modulation		GFSK (BT = 0.5, Index: 0.28 – 0.35)
Channel Space		1 MHz
Sensitivity		-70 dBm (for 0.1% BER)
Maximum Signal Level		-20 dBm
Interference Performance	$C/I_{co\text{-}channel}$	11 dB
	C/I_{1MHz}	0 dB
	C/I_{2MHz}	-30 dB
	$C/I_{>=3MHz}$	-40 dB
	C/I_{image}	-9 dB
Out-of-band Blocking	30 MHz – 2000 MHz	-10 dBm
	2000 – 2399 MHz	-27 dBm
	2498 – 3000 MHz	-27 dBm
	3000 MHz – 12.75 GHz	-10 dBm
Intermodulation Characteristics	Interference Frequency	3, 4, 5 MHz
	Interference Level	-39 dBm
	Bluetooth Signal Level	6 dB above sensitivity
RSSI	Range	-60 dBm±4 + 20±6 dB
	Accuracy	±4 dB
Radio Frequency Tolerance	Transmitted Frequency Accuracy	±75 kHz
	Frequency Drift	±25 kHz in one slot

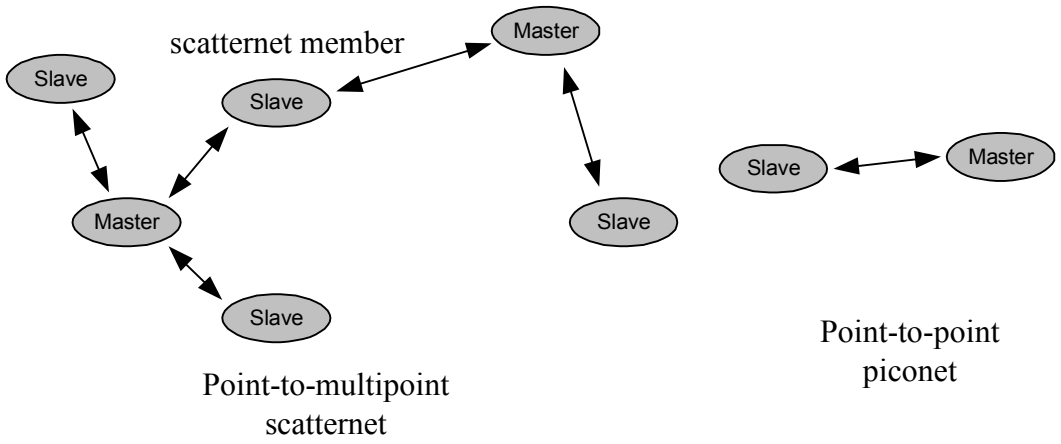


Fig. 2.2 Point-to-point and point-to-multipoint and scatternet topology in Bluetooth

2.3.2. Modulation Format

The modulation format is Gaussian Frequency Shift Keying (GFSK) with a bandwidth×bit time product (BT) of 0.5. The modulation index is between 0.28 and 0.35. For the 1Mbps data rate in Bluetooth, frequency deviation can be from ±140 to ±175kHz.

The GFSK signal is generated as follows: First the data stream d(t) is filtered using a Gaussian filter with the following impulse and frequency responses:

$$h(t) = e^{-\frac{1}{2}(\frac{t}{\tau})^2} \tag{2.1}$$

$$H(\omega) = \tau \cdot \sqrt{2\pi} e^{-\frac{1}{2}(\tau\omega)^2} \tag{2.2}$$

This is similar to the familiar e^{-x^2} shape of the Gaussian, or normal, probability density function. In the equation, ω is the frequency in rad/sec and τ is a constant. A peculiar property of the Gaussian filter is that both its frequency and impulse responses are Gaussian. Figs. 2.3 and 2.4 show the impulse and frequency responses of a Gaussian LPF, respectively. From equation (2.2), the bandwidth of the filter can be written as:

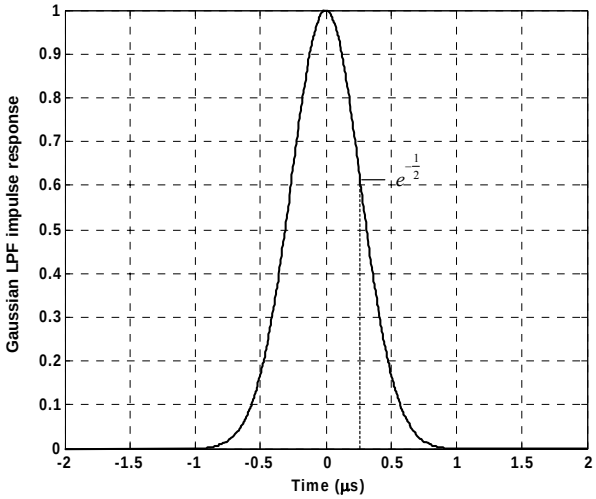


Fig. 2.3 Gaussian LPF impulse response

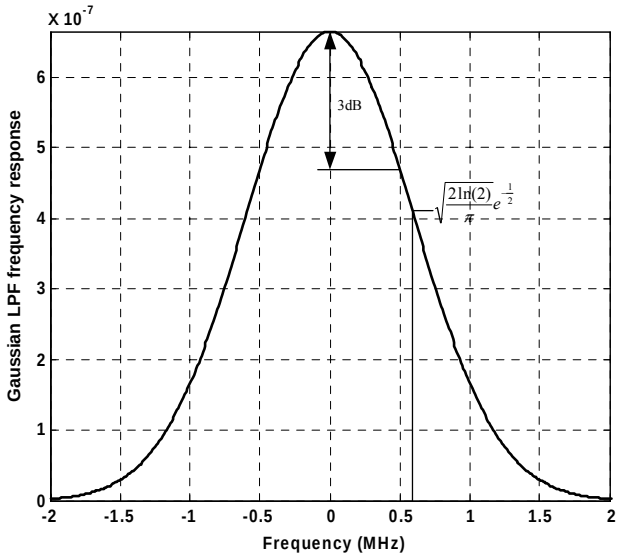


Fig. 2.4 Gaussian LPF frequency response

$$B = \frac{\sqrt{\ln(2)}}{2\pi}\frac{1}{\tau} \tag{2.3}$$

The Gaussian filter is often specified in terms of its *relative bandwidth*, relative to the bit rate, or BT:

$$BT = (Filter\ Bandwidth)\cdot(Bit\ Period) = \frac{Filter\ Bandwidth}{Bit\ Rate} = \frac{T_b}{\tau}\frac{\sqrt{\ln(2)}}{2\pi} \tag{2.4}$$

Higher BT means less intersymbol interference (ISI) but higher bandwidth. As a good compromise, BT=0.5 is specified in Bluetooth standard [1]. So the bandwidth is 500kHz. Therefore, the output of the Gaussian filter is given by:

$$gf(t) = d(t) * e^{-\frac{1}{2}(\frac{\pi}{\sqrt{\ln(2)}}\frac{t}{T_b})^2} \tag{2.5}$$

Note that both $d(t)$ and $gf(t)$ have a maximum and minimum of 1 and -1 representing 1 and 0 bits, respectively. The Gaussian filtered data $gf(t)$ is then used to frequency modulates the carrier. The output GFSK signal is expressed as:

$$gfsk(t) = \sin(2\pi(f_C + f_d \cdot gf(t)) \tag{2.6}$$

Where f_C is the center frequency and f_d is the frequency deviation. Thus the instantaneous frequency of the GFSK signal transitions between a frequency of $f_c - f_d$, representing binary 0, and $f_c + f_d$, representing binary 1. In Bluetooth standard [1], the frequency modulation index is specified between 0.28 and 0.35. The modulation index is the ratio between the frequency deviation $2 f_d$ and the bit rate. So the corresponding f_d must be between 140 and 175 kHz. Fig. 2.5 shows representative examples of the unfiltered, Gaussian filtered, and Gaussian FSK signal. For illustration purposes, the GFSK signal in Fig. 2.5 has 1MHz center frequency.

2.3.3. Frequency Band

Bluetooth operates in the 2.4GHz Industrial Scientific Medicine (ISM) band. In a vast majority of countries around the world the range of this frequency band is 2400MHz – 2483.5MHz. By utilizing Time Division Duplex (TDD), transmitter and receiver share the same frequency band. The regulators expect lots of devices to be using the same spectrum, so they have set out rules for using ISM bandwidth to make sure that devices

can share the bandwidth. The rules state that you must spread the power of your transmissions across the ISM band somehow. Two main methods are used for spreading out the power: direct sequence spread spectrum (DSSS) and frequency-hopping spread spectrum (FHSS). DSSS smears a transmission across a wide range of frequencies at low power while FHSS spectrum uses a small bandwidth but changes (or hops) frequency after each packet.

Bluetooth uses frequency-hopping spread spectrum as shown in Fig. 2.6. There are 79 channels of 1MHz each. Transmitters change frequencies 1,600 times every second. Occasionally, two piconets may collide on the same channel, but they will just hop off to new frequencies and retransmit any data that was lost. This technique also minimizes the risk that portable phones or baby monitors will disrupt Bluetooth services, since the effect of any interferer on a particular frequency will last only a tiny fraction of a second. Bluetooth uses the master's device ID to algorithmically determine the frequency hopping pattern. This algorithm generates a unique pattern that is quite random and exhibits an extremely long repeat cycle.

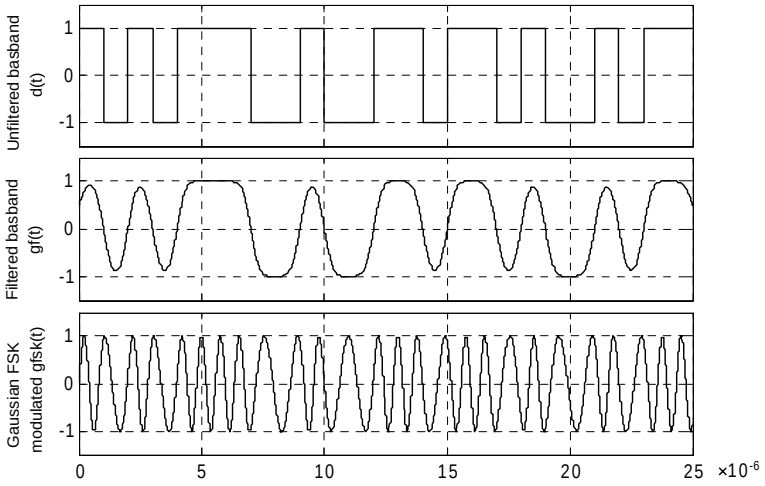


Fig. 2.5 GFSK modulation steps

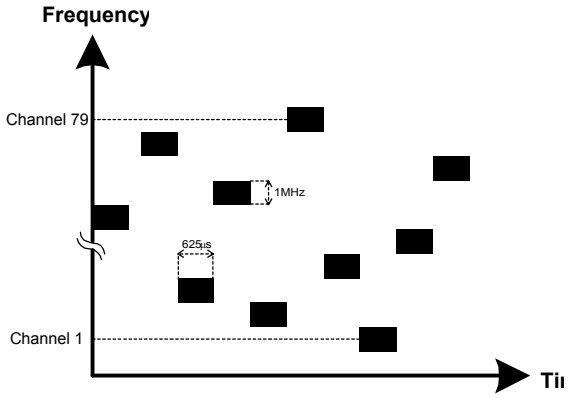


Fig. 2.6 Frequency hopping diagram

2.3.4. Blocking Requirements

The blocking test for Bluetooth is performed by applying a Bluetooth-modulated desired signal 10 dB (for Co-channel, 1 MHz and 2 MHz interference) or 3 dB (for all other frequencies interference) over the reference sensitivity level. Then a Bluetooth-modulated interfering signal is applied to the receiver at discrete increments of 1 MHz from the desired signal with a magnitude as shown in Table 1. Five spurious response frequencies are allowed at frequencies with a distance of equal or greater than 2MHz from the wanted signal. On these spurious response frequencies a relaxed interference requirement C/I = -17 dB shall be met, where C is the carrier power and I is the interference power, respectively. Usually, each channel is allowed a different set of exceptions. If a low-IF receiver is implemented, C/I degrades if I represents the image signal. Spurious response frequency may be used to relax image rejection requirement in Low-IF receivers.

2.3.5. Intermodulation Requirements

The adjacent channel immunity test is performed by applying one static sine wave signal at f_1 with a power level of –39 dBm and one Bluetooth modulated signal at f_2 with

also a power level of –39 dBm to the input of the receiver while a wanted signal 6 dB above the reference sensitivity is applied. The receiver must maintain a 0.1% BER. And be aware when performing this intermodulation test, the effects of noise in receiver channel are also there. It is necessary to make sure C/(I+N) is high enough to maintain required BER, where N is the noise floor level.

2.3.6. Sensitivity

The actual sensitivity level is defined as the input level for which a raw bit error rate (BER) of 0.1% is met. The requirement for a Bluetooth receiver is an actual sensitivity level of -70dBm or better. The power level -70dBm is defined for 50Ω impedance. This power can be written in terms of rms voltage and impedance as follows:

$$P\,(in\,dBm) = 10\log(\text{power in } mW) = 10\log(\frac{V_{rms}^{2}}{R}) + 30 \qquad (2.7)$$

So the power level -70dBm is equivalent to an rms voltage of 70.7μVrms. It is important at this point to clarify the different representations of signal power:

Power in dBm ≡ decibels relative to one milliwatt.

Power in dBW ≡ decibels relative to one watt.

Power in dBV ≡ decibels relative to one volt.

Power in dBc ≡ decibels relative to carrier signal power.

2.4. What is 802.11b?

Also known as Wi-Fi (for Wireless Fidelity), 802.11b emerged in 1999 and is the most popular wireless networking standard. Operating in the 2.4GHz radio band, 802.11b is also the current mainstay of the 802.11 family of wireless networking standards

established by the IEEE (Institute of Electrical and Electronics Engineers). 802.11 defines the PHY (physical) and MAC (media access control) layers of the protocol. All of the other layers are identical to the 802.3 (Ethernet) protocol. 802.11a was proposed before 802.11b, hence the designation in 802.11a. 802.11b, however, came to the market first.

802.11a/b uses the unlicensed spectrum for transmission and thus it must use spread spectrum techniques. This process increases the communication channels interference immunity or the processing gain, decreases interference between multiple users and increases the ability to re-use the spectrum. 802.11b uses the 2.400 GHz to 2.483 GHz spectrum. 802.11 is the wireless extension of 802.3 and supports all the underlying protocols that Ethernet uses. An Access Point (AP) is the center of the Basic Service Set (BSS) which may overlap partially, completely or not at all with each other without fear of interfering with functionality. Mobile users can roam from AP to AP and these APs are connected together with other APs using the same ESS_ID which forms an Extended Service Set (ESS). Each AP has its own MAC and IP addresses and they are fault tolerant when setup with multiple failure points. Addition of capacity to the network can be as simple as adding APs to a new Ethernet port which uses the same ESS_ID.

802.11b uses DSSS (Direct Sequence Spread Spectrum) to disperse the data frame signal over a relatively wide (30 MHz) portion of the 2.4 GHz band. This results in greater immunity to radio frequency interference as compared to narrowband signaling. Because of the relatively wide DSSS signal, you must set the 802.11b AP to specific channels to avoid channel overlap which might cause a reduction in performance. In order to actually spread the signal, the transmitter combines the Physical Layer

Convergence Procedure protocol data unit PLCP (PPDU) with a spreading sequence through the use of a binary adder. PLCP is a frame modification technique used by 802.11b which is out of the scope of discussion in this paper. For higher data rates (5Mbps, 11Mbps) 802.11b uses Complementary code keying (CCK) to provide spreading sequences.

Detailed 802.11b standard specifications can be found in [3]. A summary of Wi-Fi RF specifications is listed in Table 2.3.

Table 2.3 802.11b radio specifications

Frequency Band		2400 – 2483.5 MHz
Duplex		Time Division
Modulation		DBPSK/DQPSK/CCK
Channel Space/ number of channels		25 MHz / 3
Sensitivity (11Mbit/s)		-76 dBm (for 0.08 FER ≡ 10^{-5} BER)
Maximum Signal Level (11Mbit/s)		-10 dBm
Adjacent Channel Rejection	C/I_{25MHz}	-35 dB
Radio Frequency Tolerance		±60kHz

2.4.1. Wi-Fi Operation

Wi-Fi networks operate in two modes: *ad hoc* networks and *infrastructure* networks. Ad hoc network is usually temporary. An ad hoc network is a self-contained group of stations with no connection to a larger LAN or the Internet. It includes two or more wireless stations with no access point or connection to the rest of the world. Ad hoc networks are also called peer-to-peer networks and Independent Basic Service Sets (IBSS). Fig. 2.7 shows a simple ad hoc network.

Infrastructure networks have one or more access points, almost always connected to a wired network. Each wireless station exchanges messages and data with the access point, which relays them to other nodes on the wireless network or the wired LAN. Any network that requires a wired connection through an access point to a printer, a file server or an Internet gateway is an infrastructure network. Fig. 2.8 shows an infrastructure network [4].

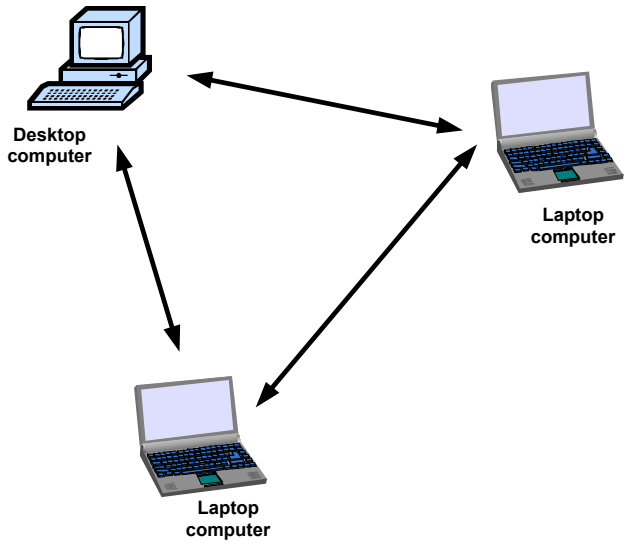


Fig. 2.7 An ad hoc wireless network with three stations using Wi-Fi

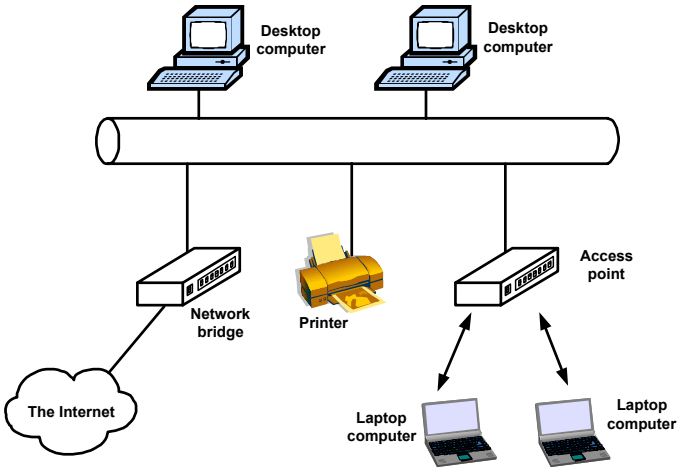


Fig. 2.8 A simple infrastructure network using Wi-Fi

2.4.2. Modulation Format

802.11b is an extension of the 802.11 standard that uses two data rates, 1 and 2Mbits/s, use DBPSK and DQPSK modulation formats, respectively. In both data rates, an 11-bit Barker sequence (+1, –1, +1, +1, –1, +1, +1, +1, –1, –1, –1) is used to spread the signal at 11MHz chipping rate. In addition to these two rates, 802.11b provides 5.5 and 11Mbit/s data rates. 8-chip complementary code keying (CCK) is employed as the modulation scheme at 11MHz chipping rate which is the same as the chipping rate in 802.11 standard, thus providing the same occupied channel bandwidth.

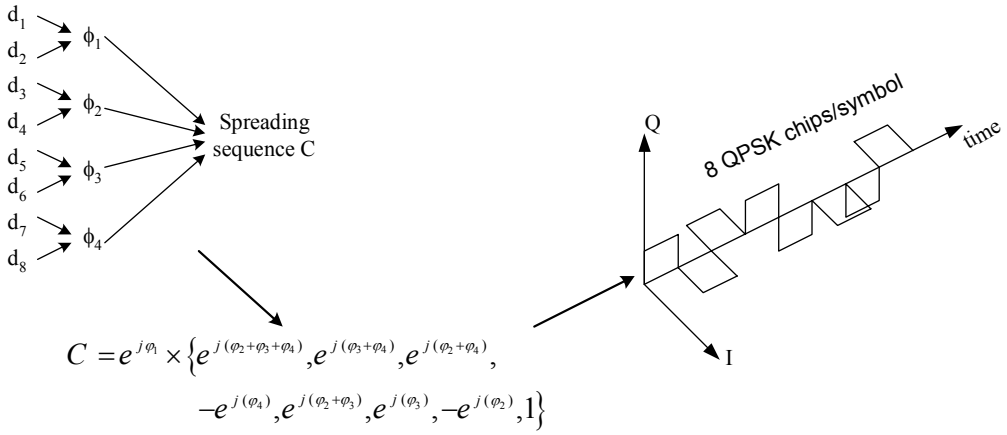


Fig. 2.9 Construction of the CCK modulated signal

CCK is a form of M-ary code word modulation where one from a set of M unique signal code words is chosen for transmission [5]. The spreading function for CCK is chosen from a set of M nearly orthogonal vectors by the data word. CCK uses one vector from a set of 64 complex (QPSK) vectors for the symbol and thereby modulates 6-bits (one-of-64) on each 8 chip spreading code symbol. Two additional bits are sent by QPSK modulating the whole code symbol and this thus modulates 8-bits onto each symbol. Fig. 2.9 illustrates how the CCK modulated signal is constructed from the 8-bits data word.

The following formula is used to derive the CCK code words and is used for spreading both the 5.5Mb/s and 11Mb/s:

$$C = e^{j\varphi_1} \times \left\{ e^{j(\varphi_2+\varphi_3+\varphi_4)}, e^{j(\varphi_3+\varphi_4)}, e^{j(\varphi_2+\varphi_4)}, -e^{j(\varphi_4)}, e^{j(\varphi_2+\varphi_3)}, e^{j(\varphi_3)}, -e^{j(\varphi_2)}, 1 \right\} \qquad (2.8)$$

Note that φ_1 is added to all code chips, φ_2 is added to all odd code chips, φ_3 is added to all odd pairs of code chips, and φ_4 is added to all odd quads of code chips. The phases φ_1, φ_2, φ_3, φ_4 are obtained in the 5.5Mb/s and 11Mb/s cases as follows:

2.4.2.1. CCK 5.5Mb/s Modulation

Four input bits, $d_0 - d_3$, are used to encode CCK phases φ_1- φ_4. The two bits $d_0 - d_1$ encode φ_1 based on DQPSK as shown in table 2.4. The data di-bits d_2 and d_3 CCK encode the basic symbol, as specified in Table 2.5. This table is derived from the formula above by setting φ_2= (d2 ×π) + π /2, φ_3 = 0, and φ_4 = d3 ×π.

Table 2.4 DQPSK encoding table

d_0 d_1	Even symbols phase change	Odd symbols phase change
00	0	π
01	π/2	-π/2
11	π	0
10	-π/2	π/2

Table 2.5 5.5Mb/s CCK encoding table

d_2 d_3	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
00	j	1	j	-1	j	1	-j	1
01	-j	-1	-j	1	j	1	-j	1
10	-j	1	-j	-1	-j	1	j	1
11	j	-1	j	1	-j	1	j	1

Table 2.6 QPSK encoding table

Di-bit pattern (d_i d_{i+1})	Phase
00	0
01	$\pi/2$
10	π
11	$-\pi/2$

2.4.2.2. CCK 11Mb/s Modulation

At 11 Mb/s, 8 bits are transmitted per symbol. The first di-bit (d_0 d_1) encodes φ_1 based on DQPSK as shown in table 2.4. The phase change of φ_1 is relative to the phase φ_1 of the preceding symbol. The data dibits (d_2, d_3), (d_4, d_5), and (d_6, d_7) encode φ_2, φ_3, and φ_4, respectively, based on QPSK as specified in Table 2.6.

2.4.3. Operating Frequency Range

Wi-Fi operates in the same frequency range 2.4-2.4385GHz as Bluetooth.

2.4.4. Blocking Requirements

Adjacent channel rejection is defined between any two channels with >25 MHz separation in each channel group. The adjacent channel rejection should be equal to or better than 35 dB using

11 Mbit/s CCK modulation for both the desired and adjacent channels. The blocking test is done at an input signal level 6 dB greater than the sensitivity level.

2.4.5. Intermodulation Requirements

There is no IM specified test for Wi-Fi. However the intermodulation requirement can be derived from the Blocking test in section 2.3.2 due to the wide band interferer. This will be discussed in details in section 5.4.

2.4.6. Sensitivity

The frame error rate (FER) is specified to be less than 0.08 at a frame length of 1024 octets (this is equivalent to about 10^{-5} BER for an input level of -76dBm. The FER is specified for the 11Mbit/s CCK modulation.

2.5. Comparing Bluetooth and Wi-Fi

Both Bluetooth and Wi-Fi operate at the same ISM frequency band. However, there are some major differences:

Speed: Bluetooth operates at a raw data rate 1Mbps, Wi-Fi at 11Mbps, a big speed difference!

Applications: Bluetooth can be considered as a cable replacement technology. It is a short-distance wireless technology having low cost and low power consumption. It is intended to be a very simple technology in which devices can communicate with each other without the need to configure hardware or drivers. Bluetooth is the choice for connecting single devices when speed is not a major issue; it is best suited to low-bandwidth applications such as sharing printers, synchronizing PDAs, using a cell phone as a modem, and (eventually) connecting appliances to one another within a 30- to 60-foot range. Wi-Fi technology on the other hand is really a wireless version of Ethernet. Widespread popularity of Ethernet makes the Wi-Fi a very viable technology because it can very easily be interfaced with any existing Ethernet setup or peripherals connected to them.

Security: Bluetooth is probably a bit more secure than Wi-Fi. For one thing, Bluetooth is designed to cover shorter distances than 802.11b. Also, Bluetooth offers two levels of

(optional) password protection. Wi-Fi has all the security risks associated with other networks: Once someone has access to one part, he or she can access the rest.

Ease of use: A single Bluetooth device can be connected to up to seven other devices at the same time. This makes it easy to find and connect to the device you are looking for or to switch between devices, such as two printers. Wi-Fi is more complex; it requires the same degree of network management as any comparable wired network.

Power: Bluetooth has a smaller power requirement than Wi-Fi, and devices can be physically smaller, making it a good choice for consumer electronics devices.

Coexistence: Bluetooth and Wi-Fi share the same band of frequencies and could, therefore, interfere with one another. Due to the modulation format, Bluetooth is more likely to interfere with Wi-Fi than vice versa [6].

Spatial capacity: Wi-Fi has a rated operating range of 100 meters in free space. In a circle with a 100-meter radius, three Wi-Fi systems can operate on a non-interfering basis, each offering a peak over-the-air speed of 11Mbit/s. The total aggregate speed of 33Mbit/s, divided by the area of the circle, yields a spatial capacity of approximately 1kbit/s per square meter. Bluetooth, on the other hand, with its low power mod has a rated 10-meter range and a peak over the air speed of 1Mbit/s. At least 10 Bluetooth piconets can operate simultaneously in the same 10-meter circle with minimal degradation, yielding an aggregate speed of 10Mbit/s. Dividing this speed by the area of the circle produces a spatial capacity of approximately 30kbit/s per square meter.

Bluetooth versus Wi-Fi RF specifications: Here are the main differences between the RF specifications of the two wireless standards.

- Operating frequency range: Bluetooth and Wi-Fi operate at the same ISM frequency band 2.4-2.483GHz.
- Channel Bandwidth: Wi-Fi bandwidth (22MHz) is much wider than Bluetooth (1MHz)
- Data rate: Wi-Fi supports 4 different data rates (1,2,5.5, 11Mbit/s) while Bluetooth has only one data rate (1Mbit/s).
- Modulation format: Bluetooth uses GFSK modulation while Wi-Fi uses DBPSK/DQPSK/CCK depending on the data rate.
- Sensitivity: Bluetooth sensitivity is -70dBm while Wi-Fi specifies a lower sensitivity level at -76dBm.
- Adjacent Channel Rejection (ACR): Bluetooth specifies ACR > 40dB at 3MHz offset while Wi-Fi has ACR > 35dB at first adjacent channel (25MHz offset).

2.6. What is UWB?

Traditional wireless systems operate within the confines of a narrow band of frequencies assigned by the government regulatory authorities. Ultrawideband is different. UWB technologies occupy a broad swath of frequencies, typically 1.5 to 4GHz wide, that covers many already assigned frequency bands in the 1 to 6 GHz range. UWB purports to occupy these frequencies without causing undue interference. It does so by emitting a power so low that it meets US Federal Communication Commission (FCC) constraints. FCC Part 15, set for incidental radiation from devices like laptops, hair dryers, and electric drills. However, UWB systems need a waiver from the FCC Part 15 rules because they function as intentional radiators. The FCC has published a notice of the proposed rule making that could lead to such a waiver.

Traditional wireless systems operate within the confines of a narrow band of frequencies assigned by government regulatory authorities. Ultrawideband is different [7], UWB technologies occupy a broad swath of frequencies, typically 1.5 to 4 GHz wide, that cover many already-assigned frequency bands in the 1 to 6 GHz range. UWB purports to occupy these frequencies without causing undue interference. It does so by emitting a power so low that it meets US Federal Communication Commission constraints, FCC Part 15, set for incidental radiation from devices like laptops, hair dryers, and electric drills. However, UWB systems need a waiver from the FCC Part 15 rules because they function as intentional radiators. The FCC has published a Notice of Proposed Rule Making [8] that could lead to such a waiver. Because of their very low radiated power, UWB systems are impractical for long-range communication use, but they appear ideal for SRW applications, particularly in the wireless PAN range of 10 meters or less. Laboratory systems have already demonstrated data bandwidths in excess of 100 Mbps over distances greater than 10 meters, with less than 200 microwatts of average radiated power—about one fifth that of a low-power Bluetooth link. Technically, a UWB system is defined as any radio system that has a bandwidth greater than 25 percent of its center frequency, or greater than 1.5 GHz. UWB technology first appeared in the 1980s, primarily for use in radar [9]. Recent advances in low-cost, low-power switching technology and processing have made it practical to consider using UWB for consumer-grade communication devices. UWB systems emit very narrow pulses with sharp rise times, with the narrowness of these pulses giving rise to UWB's broadbanded nature. Systems based on this emerging technology vary widely in their projected spatial capacity, but one UWB developer has measured peak speeds of more than 50 Mbps at a

range of 10 meters. That developer projects that at least six such systems could operate within the same 10-meter-radius circle and experience only minimal degradation [10]. Following the same calculation process, the projected spatial capacity for such a system would be more than 1,000 Kbps per square meter. Fig. 2.10 provides a side-by-side comparison of Bluetooth, Wi-Fi, and UWB spatial capacities. As shown, UWB appears to offer a substantial advantage. The Hartley-Shannon law offers a plausible reason for UWB's spatial-capacity advantage, as described in the following equation:

$$C = B\log_2(1+\frac{S}{N}) \tag{2.9}$$

Where:

C = Maximum channel capacity, in bits per second

B = Channel Bandwidth in Hertz

S = Signal power, in Watts

N = Noise power, in Watts

Because the upper bound on a channel's capacity grows linearly with the total available bandwidth, UWB systems, which occupy 1.5 GHz or more, have inherently greater headroom for expansion than more bandwidth-constrained systems. UWB technology for SRW communications is still in its earliest days. It is not yet standardized, has its own multiple competing variations, and has not received necessary regulatory approvals. Nonetheless, as a long-term target, UWB appears to have enormous potential, especially as a wireless PAN technology.

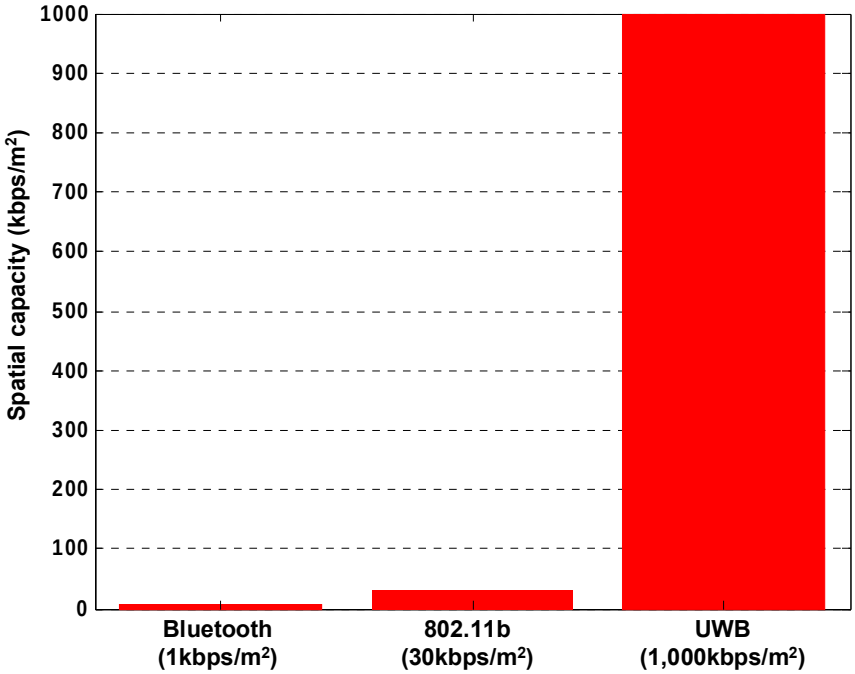


Fig. 2.10 Comparison of spatial density of Bluetooth, Wi-Fi, and UWB

CHAPTER III

BLUETOOTH RECEIVER ARCHITECTURE AND CHANNEL SELECTION FILTER

Our Bluetooth receiver design project started in the summer of 2001 with a group of 6 PhD students. We started by studying different receiver architectures [11] to achieve the highest level of integration, lowest power consumption, and best performance. Of course, all these requirements are not met in a single architecture, and therefore, tradeoffs were closely studied to find the best architecture that meets the standard specifications with enough margins at lower cost. Two very common architectures are used for Bluetooth receivers, direct-conversion and low-IF.

Although the direct-conversion architecture lends itself to higher integration levels and lower power consumption, it is plagued by quadrature demodulation phase errors, quadrature gain phase mismatch, DC offsets, 1/f noise, and LO feedthrough [12]. Low-IF architecture [13] can be used to avoid the DC offset and 1/f complications associated with direct-conversion. However, Low-IF architecture suffers from the image problem due to the non-zero IF frequency.

The choice of the most suitable receiver architecture depends on many parameters in the wireless standard (e.g. channel bandwidth, preamble time, blocking specifications, sensitivity, modulation format, etc..). In the following two sections, both possible architectures of the Bluetooth receiver will be discussed in some detail. The channel selection filter design, which I was responsible for, is described in detail in section 3.3. The receiver and the filter were fabricated in TSMC 0.35μm CMOS process.

3.1. Direct-Conversion Receiver Architecture

In direct-conversion receiver (DCR) architecture, the signal is down-converted directly from RF to baseband. A low-pass filter is then employed to suppress nearby interferers as shown in the simplified diagram in Fig. 3.1. The use of quadrature I and Q channels is necessary in the Bluetooth case because the signal is frequency-modulated, and therefore the two sidebands of the RF spectrum will carry different information. The spectrum of the complex output signal I+jQ will be a replica of the signal spectrum at RF, but down-converted around dc. Despite the simplicity of the DCR architecture, it suffers from some serious design issues that do not exist or are not as serious in low-IF receivers.

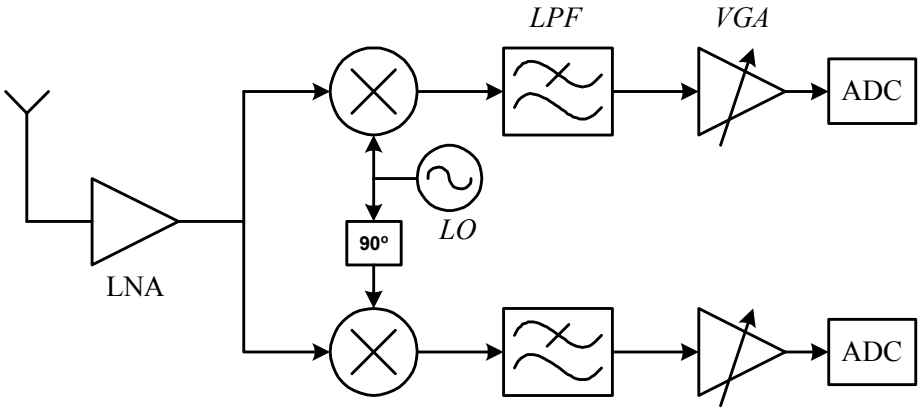


Fig. 3.1 Direct-conversion receiver architecture

3.1.1. DC Offsets

There are different sources of DC offsets in an integrated receiver; (1) components' mismatches, (2) LO self-mixing, and (3) interferers self-mixing. These sources are explained as follows:

(1) Typical MOS transistor V_T mismatches are in the order of few millivolts. This might be quite higher than the desired signal level at the mixer output.

(2) Due to capacitive and substrate coupling, isolation between the LO port and the inputs of the mixer and the LNA is finite. This effect is called LO leakage. The leakage signal appearing at the input of the LNA and the mixer is now mixed with the LO signal thus producing a dc component at the mixer output. If the LO signal level is 0dBm, and isolation between LO and LNA input is 60dB, then the LO signal at the LNA input is about -60dBm, quite substantial compared to the minimum signal (sensitivity) level at the receiver input.

(3) This effect is similar to LO self mixing. When a large interferer leaks from the LNA or mixer input to the mixer LO port, it mixes with itself and generates a low frequency beat at the mixer output corresponding to amplitude variations in the interferer. This resulting offset is even harder to reject since it is varying with time.

This means that if the desired signal level at the end of the receiver chain is at the full swing of the final stage, the dc offset generated by mismatches will saturate the receiver stages. Therefore, this dc offset has to be rejected before it gets amplified by the receiver stages. A possible approach to removing the offset is to employ ac coupling, i.e. high-pass filtering, in the down-converted signal path. However, since the spectrum of the Bluetooth GFSK signal exhibits a peak at dc, such signal may be corrupted if filtered with high cutoff frequency.

3.1.2. 1/f Noise

In modern technologies and for the minimum gate-length transistors required by RF circuits, the 1/f noise (also called flicker noise) component might exceed the white noise up to several megahertz. On the one hand, flicker noise is not a limiting effect for linear

RF circuits, as in the low noise amplifier (LNA) since the operating frequency is much higher than the corner frequency. On the other hand, since minimum length transistors are used in the switching transistors and due to the nonlinear operation of the mixer and the finite slope of the LO signal, flicker noise of the switches appears at the baseband output of the mixer. Flicker noise of the transistors used in the baseband circuits also falls in the signal band and degrades the system noise figure (NF). NF degradation depends on the flicker noise corner frequency and the channel bandwidth. In the case of Bluetooth, the -3dB bandwidth of the signal is about 500kHz, while the 1/f corner frequency is about 1MHz or even larger for smaller transistor lengths. This NF degradation might be so significant that it disqualifies DCR architecture as the optimum choice for Bluetooth. The effect of flicker noise can be reduced by a combination of techniques. As the stages following the mixer operate at relatively low frequencies, they can incorporate longer devices to minimize the magnitude of the flicker noise. Moreover, periodic offset cancellation also suppresses low-frequency noise components through correlated double sampling.

3.1.3. Even Order Distortion

Unlike other architectures, even-order distortion in the LNA and mixer input transistor becomes problematic in DCR architecture. Suppose, as shown in Fig. 3.2, two strong interferers ($A_1\cos(\omega_1 t) + A_2\cos(\omega_2 t)$) close to the desired channel experience a second order nonlinearity in the LNA represented as $y(t) = \alpha_1 x(t) + \alpha_2 x^2(t)$, then y(t) contains a low frequency term $\alpha_2 A_1 A_2 \cos(\omega_1 - \omega_2)t$. Upon multiplication by $\cos(\omega_{LO} t)$ in an ideal mixer, such a term is translated to high frequencies and hence becomes unimportant. In reality, however, mixers exhibit a finite direct feedthrough from the RF

input to the IF output due to mismatches between transistors and deviation of LO duty cycle from 50%. The natural solution to suppress even-order distortion is to use differential LNA and mixer. However, two issues arise here. First the antenna and the duplexer filter are usually single-ended. This necessitates the use of a balun (transformer) to do the single-ended to differential conversion. Baluns typically exhibit several decibels of loss at high frequencies. This loss directly raises the overall system noise figure. Second, differential LNA requires more power consumption than the single ended counterpart to achieve the same noise figure.

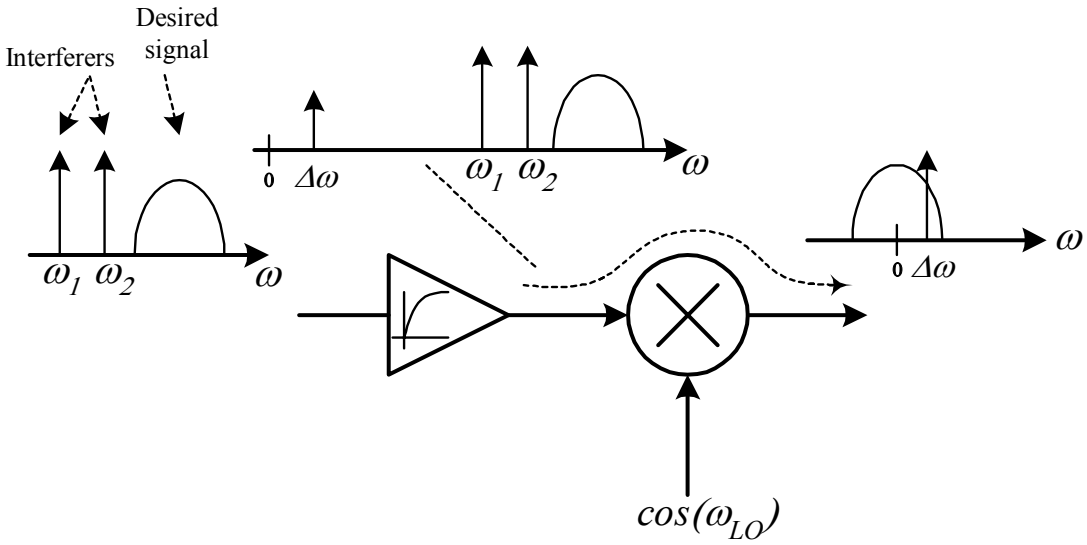


Fig. 3.2 Effect of even order distortion

3.1.4. I/Q Mismatch

Phase and magnitude mismatches between I and Q branches corrupts the downconverted signal. However, in the case of Bluetooth, since the modulation format in binary GFSK, I/Q mismatch is not a serious problem. Therefore, I/Q mismatch is much

less troublesome in DCR than in low-IF architecture where I/Q mismatch results in finite image rejection (this effect will be discussed in the next section).

3.2. Low-IF Receiver Architecture

IF receivers have been in use for a long time, and their principle of operation is very well known [14]. In an IF receiver, the wanted signal is down-converted to from its carrier to the IF by multiplying it with a single sinusoidal signal as shown in Fig. 3.3. The main disadvantage here is that apart from the wanted signal, an unwanted signal at a frequency called the image frequency (which is $2f_{IF}$ away from the wanted frequency) is down-converted to the same IF frequency. To avoid corrupting the wanted signal, the image signal must be suppressed before down-conversion by means of a band-pass RF filter. The Q of such filter is proportional to f_{RF} / f_{IF}. High Q external SAW or ceramic filters (typically 50 or more) are used for this purpose. Such filters are bulky and require impedance matching at input and output, which usually raises the power consumption that is needed in order to drive this low impedance. Furthermore, the IF frequency cannot be made arbitrarily small due to the limited Q of the external filter, and hence, raising the power consumption of the circuits operating at the IF frequency.

The image frequency problem can be mathematically explained as follows. When RF signal is multiplied by a single sinusoidal signal $\cos(\omega_{LO})$, it is equivalently multiplied by two exponentials $e^{-j\omega_{LO}}$ and $e^{j\omega_{LO}}$. Considering the signal diagram in Fig. 3.4, the spectrum of the down-converted signal is constructed by shifting the RF spectrum to the left and to the right by ω_{LO} and add the two shifted replicas. The result is two overlapping spectra, of the signal and the image, at the IF frequency.

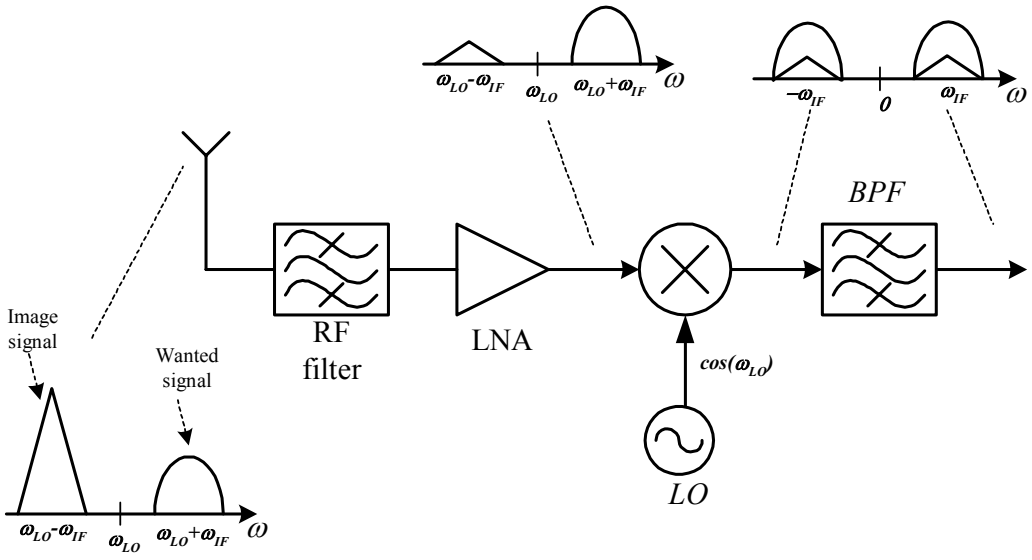


Fig. 3.3 IF receiver architecture

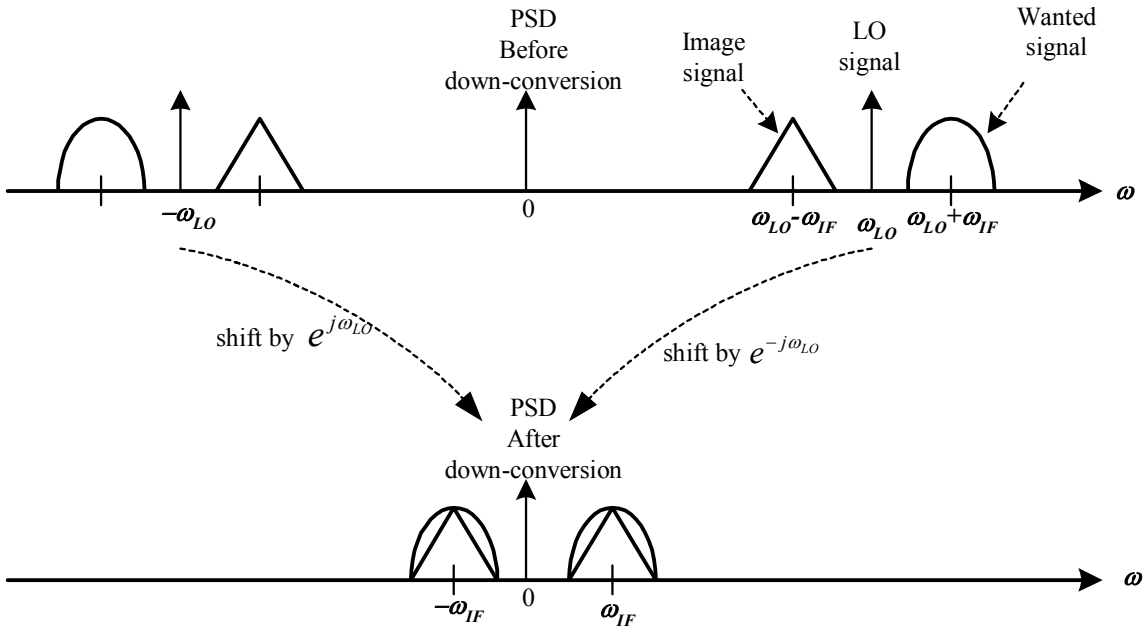


Fig. 3.4 Down-conversion with a single sinusoidal signal

An obvious solution to avoid this problem is to multiply the RF signal by only exponentials, say $e^{-j\omega_{LO}}$, which means that the down-converted signal is simply a single

shifted replica of the RF signal and therefore, no overlapping of signals spectra. The only problem now is that the signal $e^{-j\omega_{LO}}$ is a complex signal with real part ($\cos(\omega_{LO})$) and imaginary part ($-\sin(\omega_{LO})$). To implement this complex multiplication using real components, two signal branches I and Q must be constructed. In the I (in-phase or real) branch, the RF signal is multiplied by $\cos(\omega_{LO})$, while in the Q branch (quadrature-phase or imaginary), the RF signal is multiplied by $-\sin(\omega_{LO})$. Fig. 3.5 shows the signal diagram of the complex down-conversion operation. It is important to note that in each branch, the down-converted signal contains both the wanted and the image signals at the same IF frequency. However, the complex signal $I_O + jQ_O$ has the wanted signal at ω_{IF} and the image signal at $-\omega_{IF}$. The image signal can then be rejected by means of a complex filter, which will be described in detail in the following section. The same filter is also used for channel selectivity. Therefore, the Q of such filter is proportional to ω_{IF} / BW (BW is the channel bandwidth) which is small for low IF frequencies. This is the basic idea behind low-IF receiver architecture.

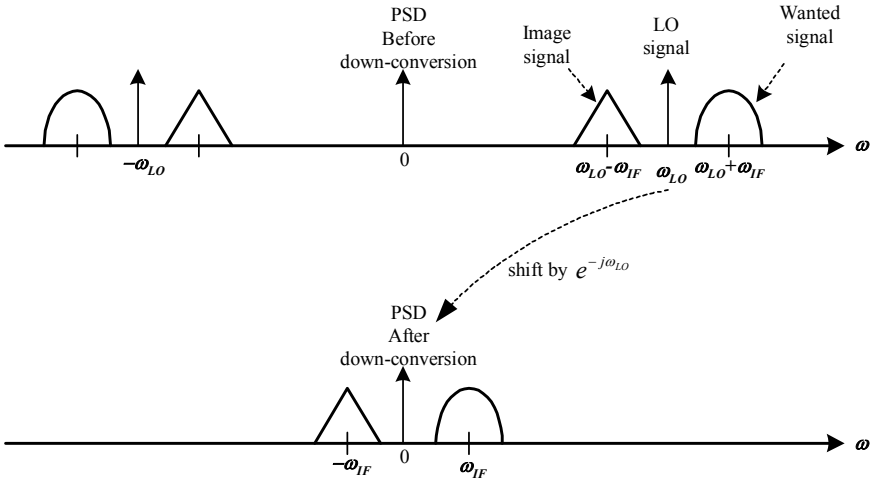


Fig. 3.5 Down-conversion with a single exponential

Fig. 3.6 shows the basic low-IF receiver architecture. Although low-IF receiver avoids the problems that exist in DCR and high-IF architectures, it has some design issues. Namely, its image rejection capability is limited by matching between I and Q branches and between the quadrature LO outputs. The effect of these mismatches is studied in the following section.

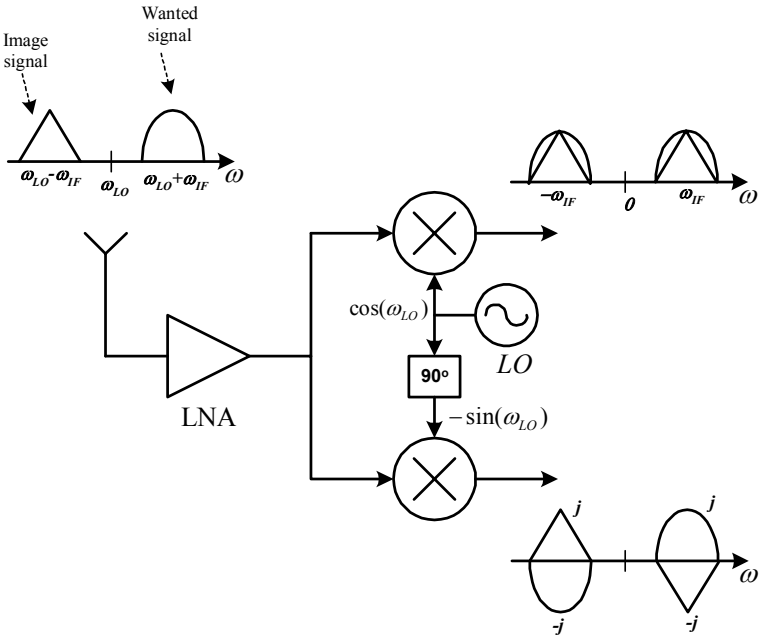


Fig. 3.6 Basic low-IF receiver architecture

3.3. Bluetooth Receiver Architecture

In a baseband Bluetooth signal, 99% of the signal power is contained within the DC to 430kHz bandwidth. Therefore, if direct-conversion architecture is used, the flicker noise and DC offset might significantly degrade the signal-to-noise ratio (SNR). Hence, a low-IF architecture seems to be a suitable architecture in Bluetooth, especially when considering the relaxed image rejection requirement in the Bluetooth standard [1]. To relax the image rejection requirement and reduce the folded-back interference level, a

very low-IF is preferable, i.e. half of the channel bandwidth. However, such a very low-IF requires a sharp cut off from the channel selection filter to reject the DC offset and flicker noise. On the other hand, a higher IF improves the demodulator performance, but the required selectivity of the channel selection filter will increase, and power consumption will be higher. As a good compromise, an IF of two times the channel bandwidth is chosen, i.e. 2 MHz. For a low-IF Bluetooth receiver, the image signal is an in-band Bluetooth modulated adjacent channel interference, which becomes co-channel interference after frequency down-conversion. It has been verified that an image rejection of 33 dB is sufficient to meet the Bluetooth specifications [15]. For the on-chip image rejection, there are several potential architectures: Hartley architecture, Weaver architecture [16], passive RC polyphase filter and active polyphase filter (also called complex filter). For the Hartley architecture, the high channel bandwidth to IF ratio makes the design of the 90^{o} phase shifter very difficult. Weaver architecture requires an extra set of mixers, a frequency synthesizer and high order band-pass filters to reject the second image; thus, the power consumption and silicon area penalty is high. Polyphase filter (also called complex filter) can be used in front of the ADC or it can be embedded in a ΣΔ ADC loop [17].Although passive RC polyphase filters can achieve a high image rejection ratio [18], they cannot achieve the required attenuation of the adjacent channel interference, especially those strong folded-back interferences, due to their limited selectivity. Extra filtering is then required to reject the adjacent channel interference, which is also true for Hartley and Weaver architectures. Another drawback of a passive RC polyphase filter is that the finite input impedance loads the RF mixers. Fortunately, an active complex filter can achieve good image rejection and adjacent channel

interference rejection. Fig. 3.7 shows one embodiment of the low-IF receiver [15]. The RF signal is amplified and down-converted to IF by the RF front end, then, the channel selection is performed by an active complex filter, which is described and proposed in this paper, and next the IF signal is passed through an amplitude limiter which removes any amplitude perturbations. As a final stage, a GFSK demodulator is employed [19].

The Bluetooth standard allows a transmitted center frequency offset as large as ±100 kHz in one time slot [1]. If the frequency offset cannot be cancelled before the channel-selection filter, the passband of the filter has to be extended to pass the desired signal with frequency offsets up to 100kHz. Since 99% of the baseband signal power is contained within 430kHz band, the complex filter passband becomes $2(430+100)=1060kHz$ centered at 2MHz.

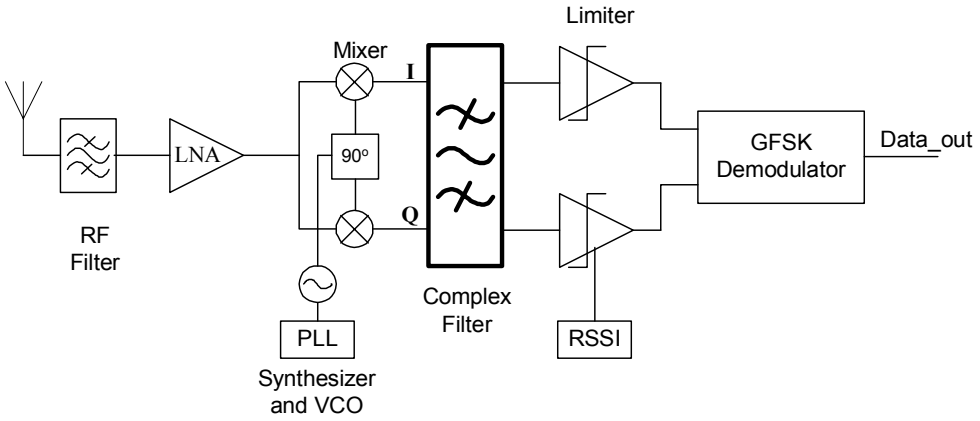


Fig. 3.7 A Low-IF Bluetooth receiver architecture [15]

Complex filter designs were found in the literature [20-23]. However, these filters either consume significant power and area (e.g. in [21], the filter draws 90mA from 5V supply and occupies an area 7.5mm2), or use a well-controlled special analog process (e.g. in [20], the process parameters are controlled within 1%). Other complex filters are

reported as part of the receiver design, and therefore, details about the filter performance were not given [23].

Partial results of the proposed complex filter have been reported in [15]. In this work, a pseudo differential OTA-C complex filter is designed and tested. A pseudo differential OTA is used to improve filter linearity and device area. A non-conventional frequency tuning scheme, simpler than conventional frequency tuning techniques, is used to compensate for process variations in a mainstream low cost TSMC 0.35μm CMOS process. This is a step forward, towards a highly integrated, low-cost Bluetooth receiver.

3.4. Channel Select Complex Filter Design

3.4.1. Complex Filter Theory

Complex filters are not new; they were invented by Sedra et al [24] in 1985. However, to justify how to implement them, a brief theoretical discussion follows. To understand the ability of complex filters to reject the image signal, consider the complex representation of the receiver block diagram shown in Fig. 3.8. For the sake of illustration, we will assume that only the desired signal and its image are present at the mixer input. Without loss of generality, we will assume the signal and the image frequencies are $\omega_{LO}+\omega_{IF}$ and $\omega_{LO}-\omega_{IF}$, respectively. After eliminating the double LO frequency term by the mixer output low-frequency pole, the result of mixing the LO and RF signals in the complex domain is:

$$B = G_{mixer}(x_{sig}e^{j\omega_{IF}t} + x_{image}e^{-j\omega_{IF}t}) = B_I + jB_Q \tag{3.1}$$

Where B_I and B_Q are the real and imaginary parts of the mixer output and can be expressed[1] as:

$$\begin{aligned} B_I &= G_{mixer}(x_{sig}\cos(\omega_{IF}t) + x_{image}\cos(\omega_{IF}t)) \\ B_Q &= G_{mixer}(x_{sig}\sin(\omega_{IF}t) - x_{image}\sin(\omega_{IF}t)) \end{aligned} \tag{3.2}$$

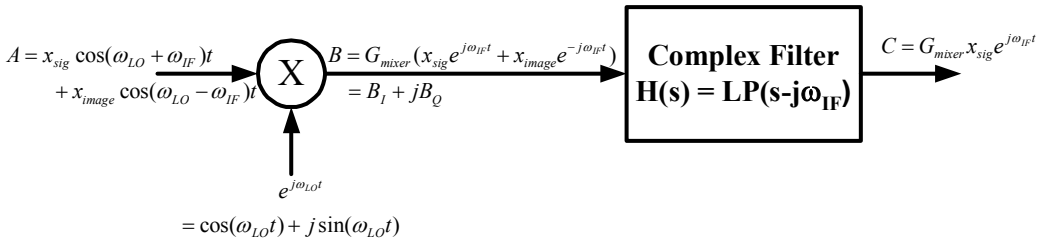


Fig. 3.8 Receiver image rejection architecture in the complex domain

Note that in the above equation the desired (image) signal in the I branch leads (lags) the Q branch by 90º. Fig. 3.9 illustrates the complex mixing operation on the desired signal and its image. Note that after down-conversion, the $2\omega_{IF}$ frequency separation between the signal and the image is still preserved. The complex channel select filter [24] is then a frequency-shifted version of a low pass filter response. This means that the filter can pass the signal at $\omega=\omega_{IF}$, while attenuating the signal at $\omega=-\omega_{IF}$. Since the filter has unsymmetrical frequency response around the $j\omega$ axis, its time domain response is complex[2] (here comes the name complex filter). However, the complex filter frequency response is symmetrical around the ω_{IF}. Since the blocking specifications of a receiver are symmetrical around the desired signal frequency, this is considered an advantage of complex filter over real BPF that has unsymmetrical frequency response around its center frequency.

[1] Note that the notations I and Q in Fig. 3.1 correspond to B_I and B_Q, respectively in Fig. 3.2.
[2] In real filters, complex poles are always conjugate, but in complex filters one single complex pole is possible.

These complex operations are practically performed as follows. Multiplication of the real RF signal by $e^{j\omega_{LO}}$ is practically performed using a quadrature mixer, which basically consists of two mixers whose LO inputs are in quadrature phase, as shown in Fig. 3.10. In the complex signal representation in Fig. 3.8, the desired signal at the mixer output is located at the positive IF frequency while the image signal is located at the negative IF frequency. In the real implementation in Fig. 3.10, the desired (image) signal in the I branch leads (lags) the Q branch by 90°. Phase and gain imbalances at the mixer output, due to LO and mixer mismatches, will cause the image signal at $-\omega_{IF}$ to spill over the image band at ω_{IF}. As a result the image rejection ratio (IRR) will be limited by these mismatches. It can be shown that the rejection limit (in dB) is given by:

$$\mathrm{IRR}_{\max}(\mathrm{indB}) = 10\log(\sin^2(\frac{\theta}{2}) + (\frac{\Delta}{2})^2\cos^2(\frac{\theta}{2}))) \approx 10\log(\sin^2(\frac{\theta}{2}) + (\frac{\Delta}{2})^2) \qquad (3.3)$$

Where Δ and θ are the gain and phase imbalances, respectively. For example, for IRR > 30dB, the maximum tolerable phase and gain mismatches (assuming equal contribution to $\mathrm{IRR}_{\max}$) are 2.5° and 4.2%, respectively. See [25] also about mismatch effects on complex filters.

The complex filter, in turn, is able to make the distinction between the signal and the image based on the phase difference between the I and Q branches. In the complex domain, the complex BPF is a frequency-shifted version of an LPF. To convert an arbitrary LPF to a complex BPF centered at ω_{IF}, every frequency dependent element in the LPF [24] should be altered to be a function of s- $j\omega_{IF}$ instead of s. The basic frequency dependent element in a filter is the integrator. Consider the simple case of converting a first order LPF with cut-off frequency ω_{LP}, to a complex filter BPF centered

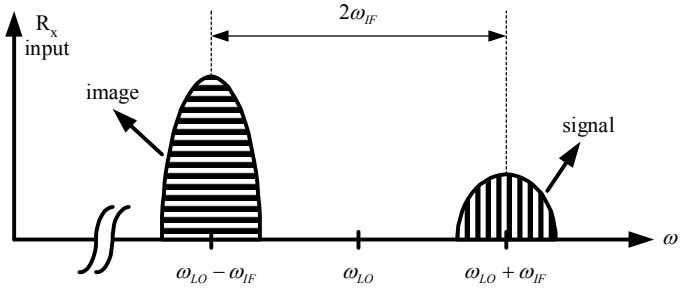
at ω_{IF}. The LPF response is shifted in frequency by placing it in a complex feedback loop as shown in Fig. 3.11(a). The complex input-output relation is given by:

$$x_o = \frac{\omega_o}{s + \omega_{LP} - j\omega_{IF}} x_i \tag{3.4}$$

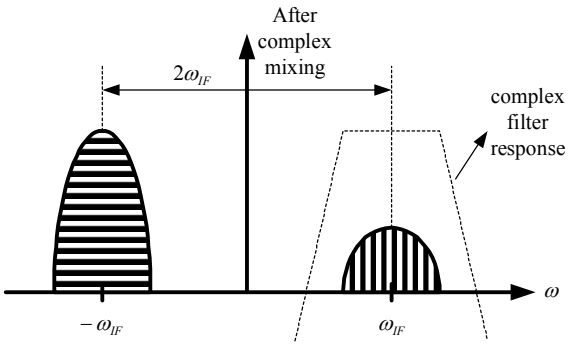
where $x_i = x_{iI} + jx_{iQ}$ and $x_o = x_{oI} + jx_{oQ}$, then from (3.4):

$$x_{oI} = \frac{\omega_o}{s + \omega_{LP}} (x_{iI} - \frac{\omega_{IF}}{\omega_o} x_{oQ}) \tag{3.5.a}$$

$$x_{oQ} = \frac{\omega_o}{s + \omega_{LP}} (x_{iQ} + \frac{\omega_{IF}}{\omega_o} x_{oI}) \tag{3.5.b}$$



(a)



(b)

Fig. 3.9 Frequency translation of a complex (quadrature) mixer (a) before complex mixing (signal A in Fig. 3.8) (b) after complex mixing (signal B in Fig. 3.8)

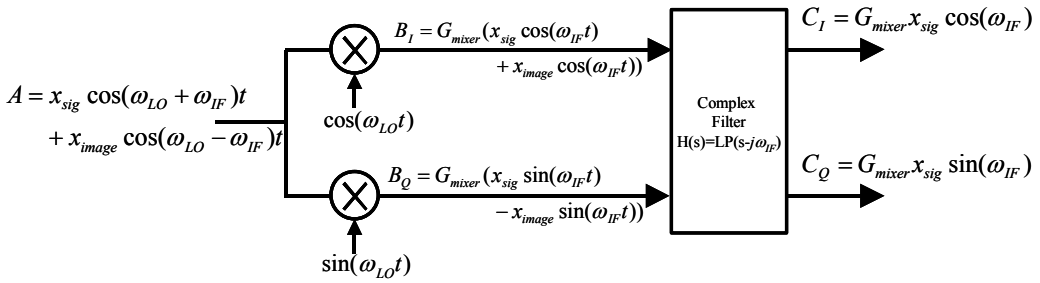


Fig. 3.10 Practical implementation of the receiver image rejection

Equation (3.5) is implemented as shown in Fig. 3.11(b). For illustration purposes, an active-RC implementation of this first order complex filter is shown in Fig. 3.11(c). Note that an inverting amplifier is needed in the cross feedback from Q branch to I branch. If a differential implementation is used, this extra inverting amplifier can be avoided by exchanging the differential signals. The corresponding pole locus of the prototype and the complex BPF are shown in Fig. 3.12. For the special case of an integrator prototype, when $\omega_{LP} = 0$, the above transformation still holds and the integrator response will be shifted to ω_{IF}. If the above transformation is applied to every frequency dependent element in the LPF prototype, the entire LPF frequency response will be shifted to ω_{IF}. Fully differential active-RC implementation of the first order complex filter is illustrated in Fig. 3.13. Note that the inverting amplifier is eliminated in this case.

3.4.2. Complex Filter Implementation

3.4.2.1. Filter Approximation

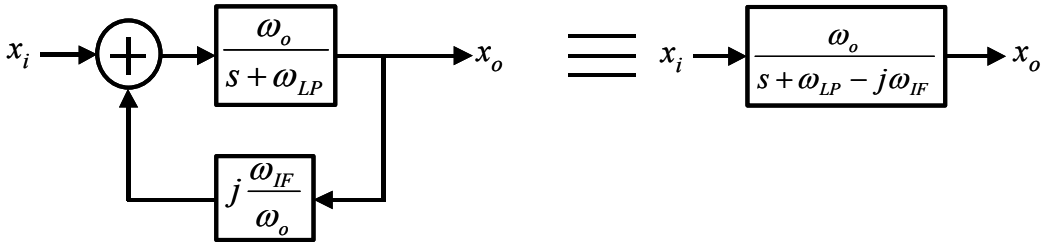
In OTA-C filters, the lowpass to complex filter transformation is done by replacing each pair of integrators, in I and Q branches, by the circuit shown in Fig. 3.14 for an OTA-C filter. Differential OTA-C implementation of the linear frequency translation is depicted in Fig. 3.15. System level simulations show that a complex filter based on a 4th

order Chebychev LPF or 6th order Butterworth LPF may be sufficient to achieve the required selectivity. The Butterworth approximation is preferred for two reasons. First, it has small group delay variation (0.6 μs) within the pass band. Second, since all the poles have the same angular frequency in a Butterworth filter, the cross-coupled OTA's will have the same transconductance value in the entire filter resulting in better matching between filter stages and between the filter and the frequency tuning circuit. The highest quality factor (Q_p) in the LPF prototype is 2. This small Q_p can tolerate process mismatches without affecting the filter performance significantly, thus it can be realized easily without using Q_p tuning. However, a frequency tuning circuit is required to compensate for the expected ω_o variations due to the process technology variations. To simplify the LPF to bandpass filter (BPF) transformation, the LPF prototype should have only grounded capacitors. If the LPF prototype was to include floating capacitors, two pairs of cross-coupled OTA's would be used for each floating capacitor (one cross-coupled pair for each capacitor terminal) for frequency shifting. Therefore, floating capacitors in the LPF prototype means increased area and power of the complex filter. The LPF prototype is implemented using three biquads.

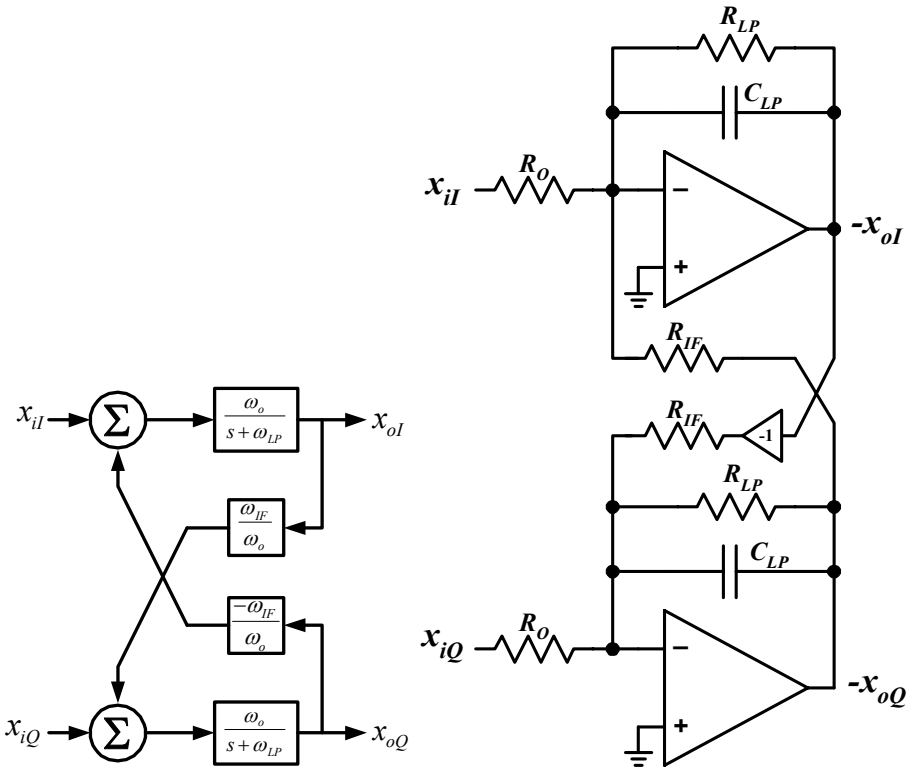
3.4.2.2. OTA Topology and CMFB

In order to reduce the input referred noise, the least number of transistors is used in the OTA, as shown in Fig. 3.16. This OTA was preferred over Nauta's transconductor [26] for two reasons. First, Nauta's transconductor has some circuitry to ensure CM stability which may not be needed in some cases, and therefore, may consume more power. Second, Nauta's transconductor is tuned through its supply voltage, and therefore, a buffer with high current driving capability and low output resistance is needed to drive

the OTA supply node. The fact that both NMOS and PMOS transistors contribute to the transconductance in Nauta's transconductor is not really relevant in this case since a low transconductance value is used.



(a)



(b) (c)

Fig. 3.11 LPF shifted to ω_{IF} (a) conceptual complex representation (b) actual building block implementation (c) Active-RC implementation

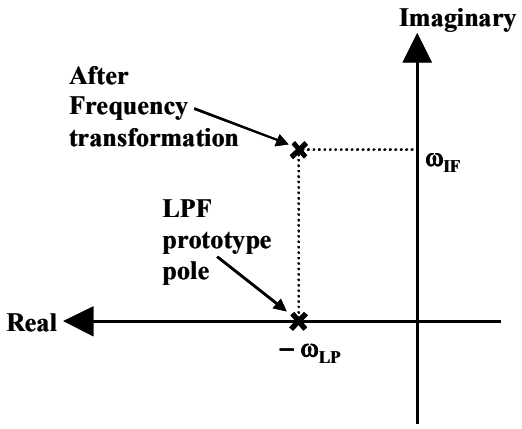


Fig. 3.12 Pole locus of the LPF prototype and the complex BPF

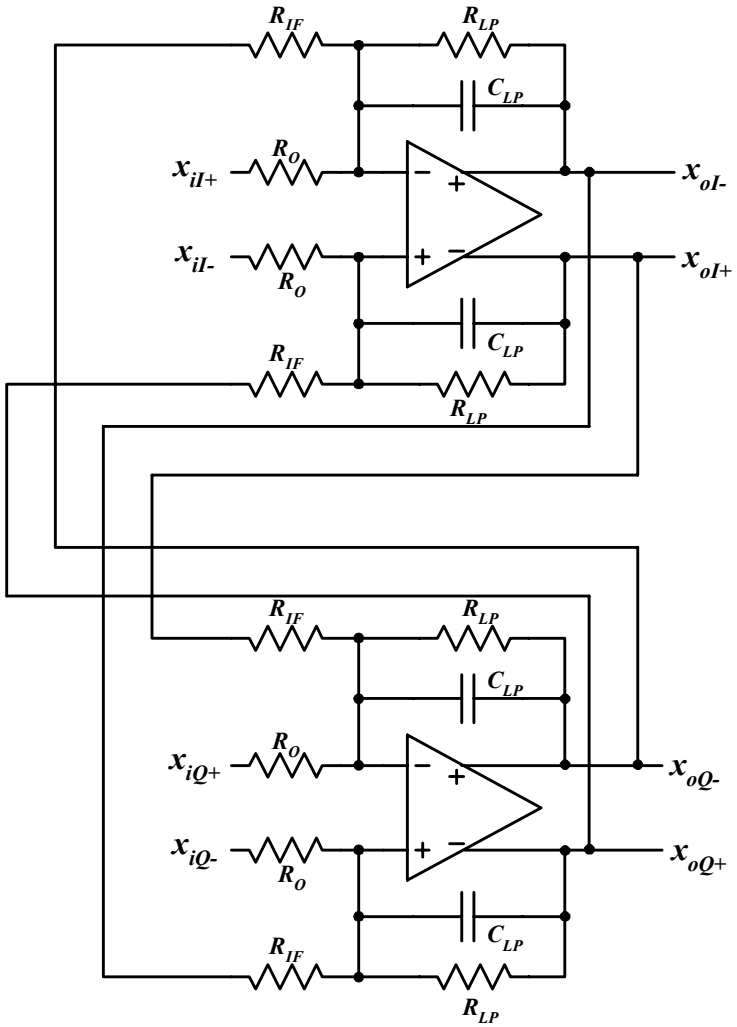


Fig. 3.13 Fully differential active-RC complex first-order LPF

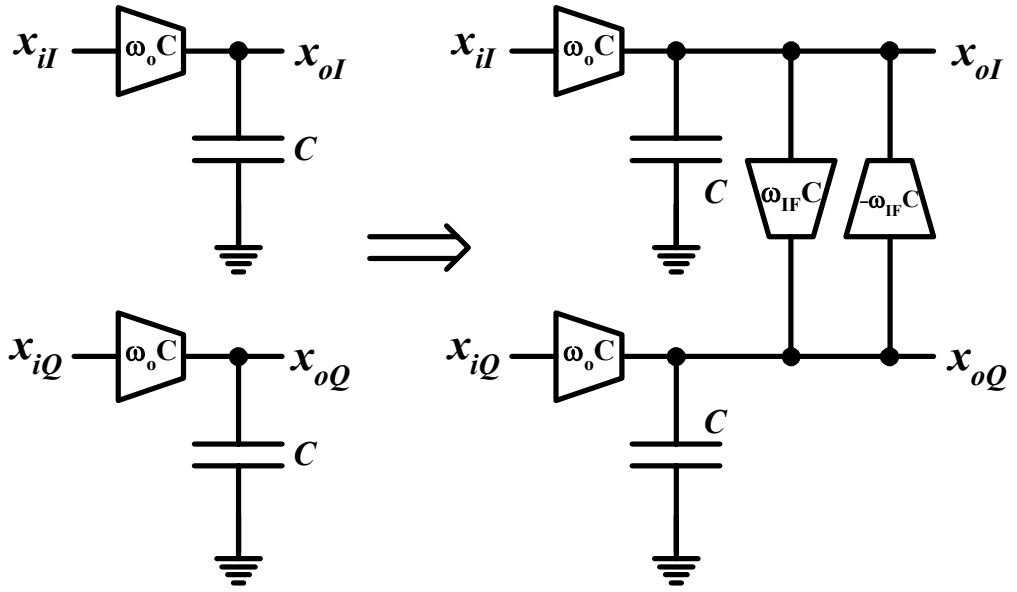


Fig. 3.14 Linear frequency translation to convert LPF to complex BPF

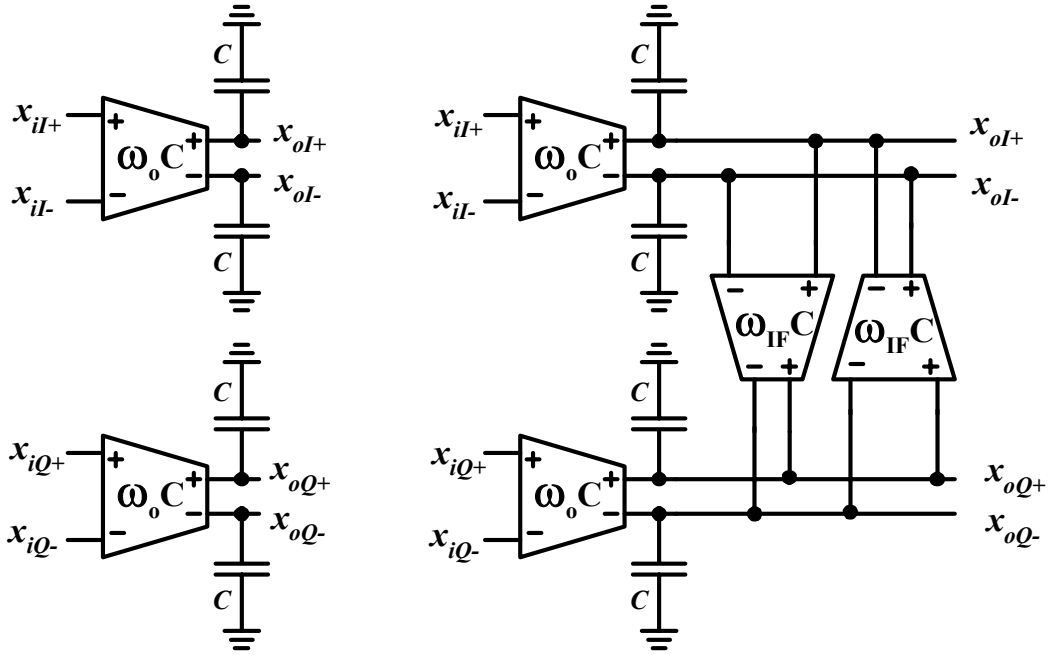


Fig. 3.15 Differential OTA-C implementation of linear frequency translation

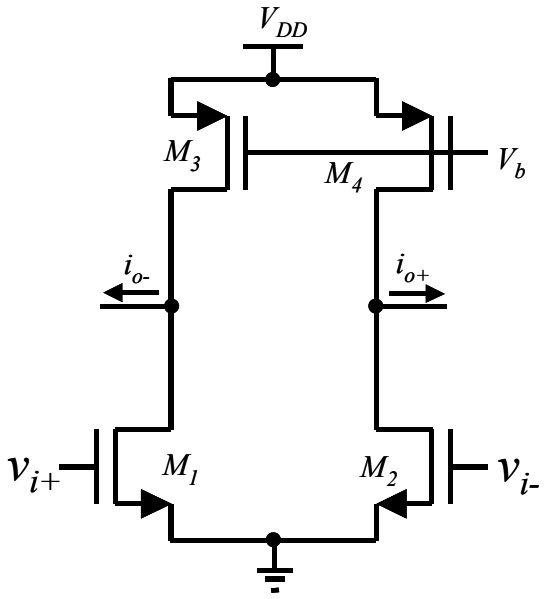


Fig. 3.16 Pseudo differential OTA

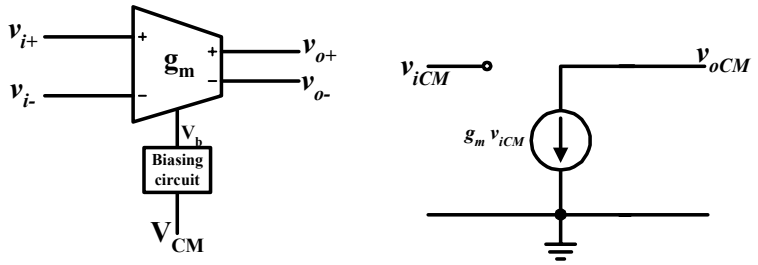
Long channel transistors (6 μm) are used in the OTA in Fig. 3.16 to enhance the output resistance, improve matching, and reduce flicker noise. A pseudo differential architecture is used to reduce the required supply voltage. The signal swing is determined by the overdrive voltage (V_{dsatn}) of the input transistors M_1 and M_2. The minimum and maximum acceptable single-ended input levels to the OTA are V_{Tn} and $V_{Tn}+2V_{dsatn}$, respectively. Therefore, the minimum power supply voltage is $V_{Tn} + 2V_{dsatn} + V_{dsatp}$. The transconductance of the OTA is linearly proportional to the input common mode (CM) voltage V_{CM}:

$$g_m = K_n' \frac{W}{L}(V_{CM} - V_{Tn}) \tag{3.6}$$

If V_b is a fixed bias voltage (i.e. no CM control), the common mode rejection ratio (CMRR) of the OTA equals unity. Therefore, V_b must be controlled using the CM input or output signal of the OTA to improve its CMRR.

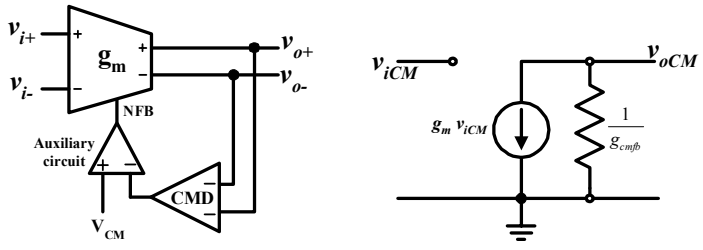
3.4.2.3. CMFF and CMFB Interconnection Strategy

Here we discuss an alternative approach to avoid the use of conventional CMFB circuits which often can be area and power hungry. Fig. 3.17(a) shows the CM equivalent of an OTA with no CM control. Note that input CM signal is transferred to the output through the CM transconductance, which happens to be the same value as given in (6). Unless the CM impedance at the output node is low enough, this biasing approach provides high CM voltage gain and may cause CM instability. The biasing circuit of the OTA in Fig. 3.17(a) is shown in Fig. 3.18(a). V_{CM} in the biasing circuit is a fixed bias voltage and is not dependent on CM input or output signals of the OTA.



OTA without CM control **CM equivalent circuit**

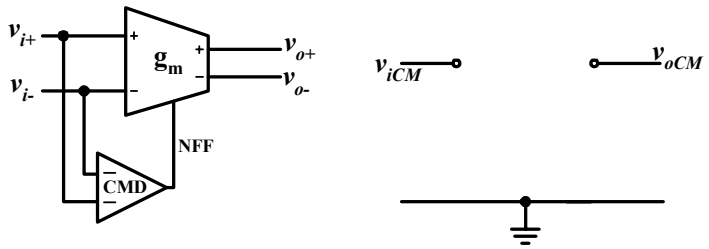
(a)



CMFB **CM equivalent circuit**

(b)

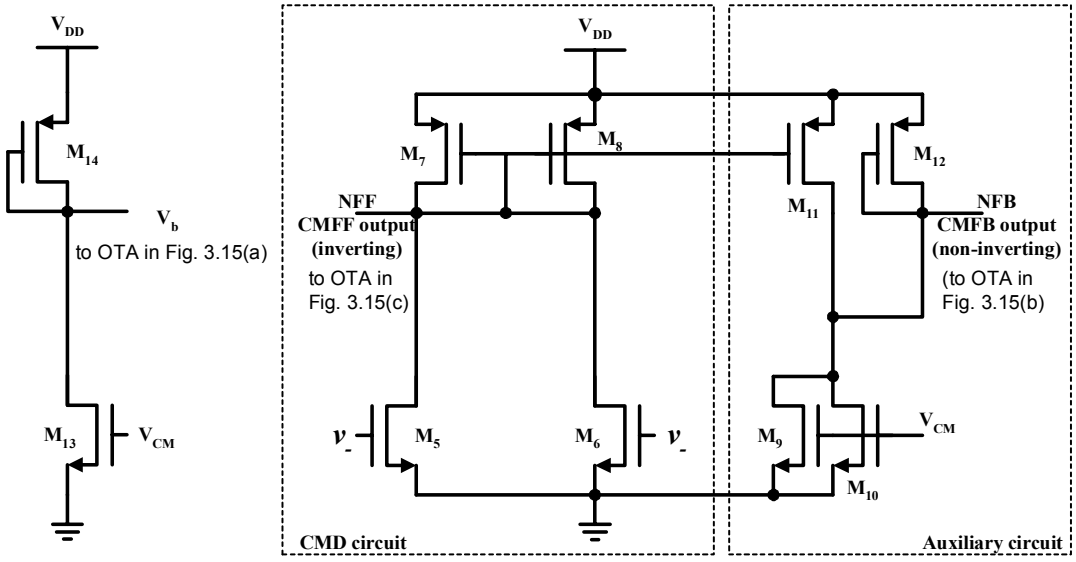
Fig. 3.17 (a) No CM control (b) CMFB circuit (c) CMFF circuit



CMFF **CM equivalent circuit**

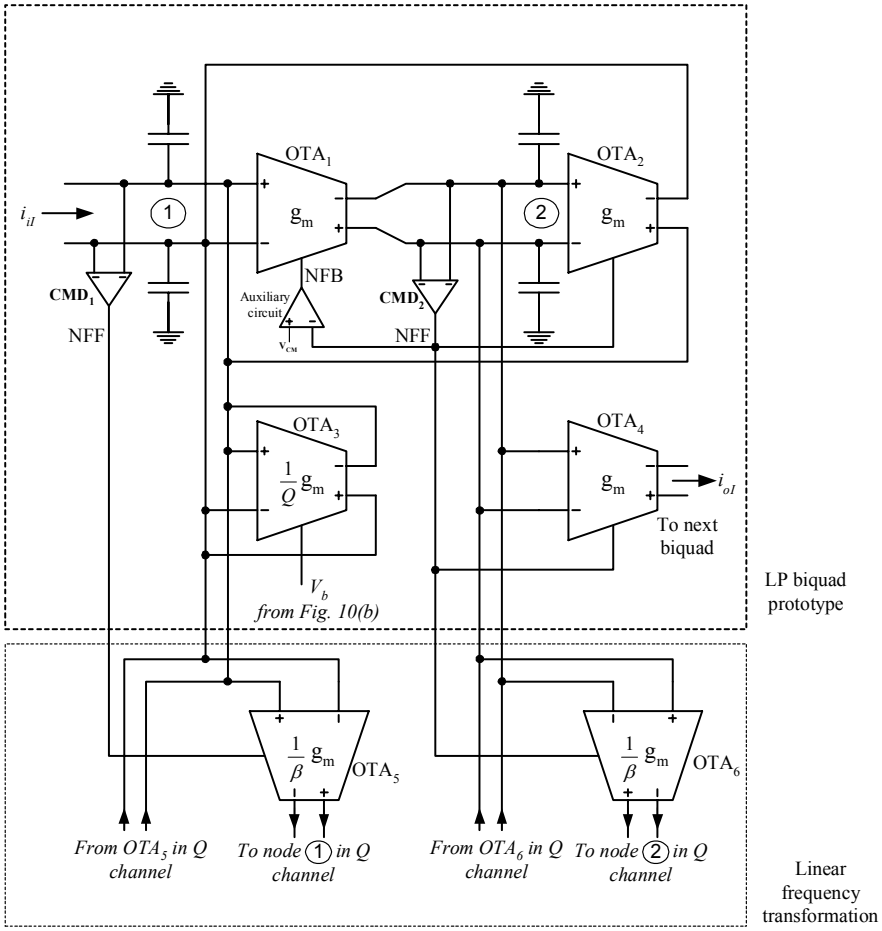
(c)

Fig. 3.17 Continued

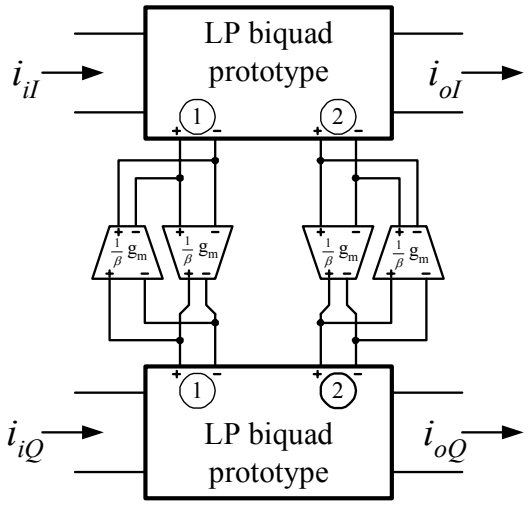


(a) (b)

Fig. 3.18 (a) Biasing circuit (b) The CMD and auxiliary circuit required for CMFF and CMFB



(a)



(b)

Fig. 3.19 (a) I branch of the complex biquadratic section (b) Conceptual complex biquad

To enhance the CMRR of the OTA's, a CMFB or a common-mode feed forward (CMFF) is used. If the output CM impedance is high, then CMFB is needed to lower this impedance and to fix the DC operating point. This is illustrated in Fig. 3.17(b) where the output CM impedance becomes $1/g_{mfb}$, where g_{mfb} is the transconductance of the CMFB loop. In Fig. 3.17(b), the common mode detector (CMD) senses the CM signal at the output node and feeds the correction signal to the bias voltage V_b of the OTA. To ensure CM stability, the magnitude of the closed loop CM gain (G_{CM}) should be less than one. For a load resistance of R_L, the G_{CM} gain of the circuit if Fig. 3.17(b), when the OTA of Fig. 3.16 is used, can be expressed as:

$$G_{CM} = \frac{-g_{mn}R_L}{1+\alpha g_{mp}R_L} \tag{3.7}$$

Where g_{mn} and g_{mp} are the transconductances M_1 and M_3, respectively, in Fig. 3.15, and α is the voltage gain of the CMD. Since the CMFB is used in cases when the load resistance is high, we can assume that $\alpha g_{mp}R_L >> 1$. In this case, the CM stability condition reduces to the following condition:

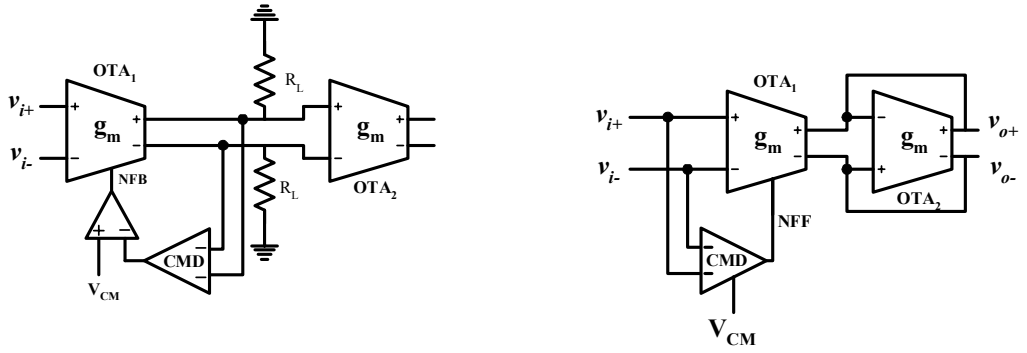
$$\alpha g_{mp} > g_{mn} \tag{3.8}$$

On the other hand, if the output CM impedance is sufficiently small, CMFB is not needed and CMFF is used to isolate the input and output CM signal of the OTA by canceling the common-mode signal. This is illustrated in Fig. 3.17(c). The polarity of the OTA indicated in Fig. 3.16 is only valid in the differential mode (DM) sense. DM transconductance polarity can be changed by just exchanging the output terminals or input terminals without adding any extra components. However, the CM transconductance does not change by exchanging the output or input terminals. In fact,

the CM transconductance of the OTA in Fig. 3.16 is always negative. Thus, a loop can be stable in DM but unstable in CM. Note also that the CMD circuit has to be an inverting type when used in CMFF to cancel the CM signal at the OTA output (see Fig. 3.17(b)), and non-inverting when used in CMFB to have negative CMFB loop (Fig. 3.17(c)). The CMD circuit is illustrated in Fig. 3.18(b). All NMOS (and PMOS) transistors are matched. The inverting output of the CMD is inverted using the auxiliary circuit to generate the non-inverting output. If the CMD is used only for CMFF, the auxiliary circuit is eliminated.

Fig. 3.19(a) shows the I branch of one of the filter biquads. OTA_4 provides the LP output current for the next current-mode filter stage. OTA_5 and OTA_6 play the same role as R_{IF} in Fig. 3.11(c) or $\omega_c C$ in Fig. 3.14. OTA_1 and OTA_2 form a negative feedback DM loop, but a positive feedback CM loop. The output node of OTA_2, node 1 is a low impedance ($1/Qg_m$) node due to the resistive connected OTA_3. Hence, no CMFB is needed at this node and only CMFF is used in all the OTA's that feed this node, excluding OTA_3. If CMFF is used in OTA_3, the CM impedance at node 1 will be very high. Instead, the bias voltage of OTA_3 is connected to a fixed voltage, independent of input and output CM voltages. The use of CMFF in OTA_2 breaks the CM loop formed by OTA_1 and OTA_2. Without CMFB, node 2 is a CM high impedance node, and hence, needs CMFB to stabilize it. CMFB loop is formed in OTA_1 through CMD_2. CMFF is also used in OTA_4, OTA_5, and OTA_6 to isolate the CM signals in this biquad stage from the next biquad and from the corresponding biquad in the Q branch. Only two common mode detectors are needed to form the CM control circuit in this biquad stage with six OTAs. By using the minimum number of CM control circuits, this efficient scheme saves

considerable power and silicon area, and contributes less noise than using a conventional CM control circuit for each OTA, as in Nauta's transconductor [26]. The proposed CM control scheme roughly consumes only about 1/3 of the area and power of CM control circuitry in conventional schemes. A CMRR in excess of 50 dB is obtained. Fig. 3.19(b) shows a block diagram for the complex biquad, which consists of two LP biquads, and two cross-coupled OTA's for each on the internal nodes 1 and 2.



(a) (b)

Fig. 3.20 (a) Circuit setup for HD3 analysis (b) Another scenario for HD3 analysis

3.4.2.4. Harmonic Distortion

Due to the long channel used in the OTA, square law characteristics can be assumed for the MOS transistors. Therefore, it can be shown that while CMFF does not introduce 3rd order harmonics, the CMFB does. To show how CMFB introduces 3rd order harmonics, consider the simplified case shown in Fig. 3.20(a). OTA1 is loaded with a linear resistor R_L to study the effect of CMFB non-linearity by itself. The CMFB on OTA_1 generates a CM 2nd order harmonic at the output of OTA_1. This harmonic mixes with the DM fundamental output of OTA_1 due to the 2nd order non-linearity of the input

transistors in OTA_2. As a result, 3rd order harmonics appear at the output of OTA_2, and the harmonic distortion can be expressed as:

$$HD3 = \frac{g_m R_L (g_m R_L - 1)}{32} \left(\frac{V_P}{V_C - V_{Tn}} \right)^2 \tag{3.9}$$

Where V_P is the peak voltage of the input signal. Two remarks are drawn in the above expression: first, the HD3 is inversely proportional to the squared overdrive voltage. Second, the HD3 due to CMFB vanishes when $R_L = 0$ or $R_L = 1/g_m$. However, in these two cases, CMFB is actually not needed. For $g_m R_L >> 1$, the above expression reduces to:

$$HD3 \approx \frac{1}{32} \left(\frac{g_m R_L V_P}{V_C - V_{Tn}} \right)^2 = \frac{1}{32} \left(\frac{V_{Pout1}}{V_C - V_{Tn}} \right)^2 \tag{3.10}$$

Where V_{Pout1} is the signal peak at the output of OTA_1. Another practical scenario that may generate HD3 is shown in Fig. 3.20(b). In this case, OTA_1 is loaded with the resistive connected OTA_2.CMFF is used for OTA_1, while OTA_2 has no CM control. The HD3 at the output is expressed as:

$$HD3 = \frac{1}{8} \left(\frac{g_{m1} / g_{m2}}{V_C - V_{Tn}} \right)^2 V_P^2 = \frac{1}{8} \left(\frac{V_{Pout1}}{V_C - V_{Tn}} \right)^2 \tag{3.11}$$

Where V_P and V_{Pout1} are the peaks of the signals at the input and output of OTA_1, respectively. Note that in the later scenario, for the same biasing conditions, HD3 is 12dB worse than the 1st scenario.

3.4.2.5. Filter Architecture

Fig. 3.21 shows the block diagram of the entire complex filter. Passive input high pass RC filters are used to isolate the CM mixer output from the filter CM input. The

voltage V_{T1} applied through the R of the high pass filter tunes the transconductance of the filter input stage, which uses the same OTA architecture shown in Fig. 3.16. An important design issue is how to distribute the gain among the filter stages. If all the gain (15 dB) is used at the filter input stage, the noise performance will be optimized but the linearity is degraded and vice versa if the gain stage is placed at the end of the filter stages. Due to the tough noise requirement on the filter, a 15 dB gain stage is placed at the filter input as shown in Fig. 3.21. Since Bluetooth uses a frequency modulation format (GFSK), in-band linearity is not a major issue. In contrast, the filter design should be focused to improve the out-of-band linearity. Since the out-of-band blockers will be attenuated by the filter, harmonics generated by the out-of-band blockers are dominated by the filter's first gain stage. Hence, to improve the overall filter linearity, the gain stage is designed to have better linearity than the filter by using larger overdrive voltage ($V_{GS} - V_{Tn}$) of the input NMOS transistors M_1 and M_2 shown in Fig. 3.16.

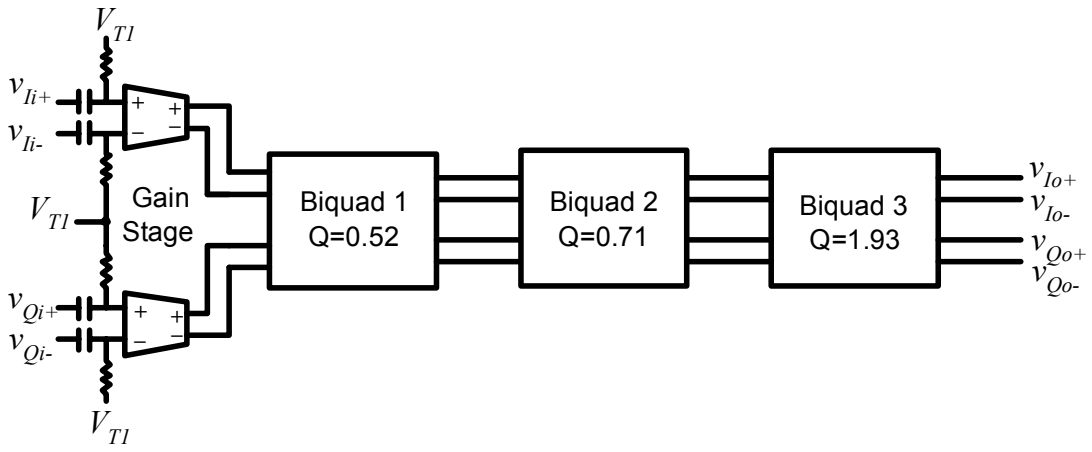


Fig. 3.21 The complete 12th order complex filter

3.4.3. Frequency Tuning Scheme

3.4.3.1. System Architecture

Fig. 3.22 shows the frequency tuning circuit of the complex filter, which is built to compensate for process variations. It consists of a relaxation oscillator, two counters to measure the oscillator and reference frequencies, a comparator, an up-down counter, and a simple D/A converter (DAC). The relaxation oscillator, discussed in the next subsection, is based on the same OTA architecture used in the filter. Under nominal conditions, the frequency of the relaxation oscillator is equal to the reference frequency (1MHz). The operation of the tuning circuit is described as follows: after system reset, the 7-b reference and oscillator counters start counting until the reference counter reaches 64. At this time, the Up/Dn counter is clocked to count up or down, or freezes according to the output of the digital comparator, which compares the content of the oscillator counter with $D_{ref} = 64$. The content of the 7-b Up/Dn counter is then converted to an analog voltage V_c (via a 7 bit DAC) to control the frequency of the oscillator (by controlling the value of g_m though V_b as shown in Fig. 3.16). When the reference counter overflows (reaches 128), it sends a reset signal to the oscillator counter to begin a new frequency comparison cycle based on the updated oscillator frequency. Eventually, the oscillator frequency will reach the reference frequency (the reference frequency is 1MHz and is derived from the 16MHz crystal oscillator used for the receiver chip) within an error depending on the DAC resolution. The same control voltage V_c is applied to V_{CM} in the filter biasing circuits and CM control circuits (Fig. 3.18) to tune the frequency to the correct value. A dead zone, depicted in Fig. 3.22, is added to the comparator transfer characteristic to avoid oscillation in the loop around the desired frequency. The width of

the dead zone is 3 counter steps around the middle count. The maximum error in the frequency-tuning loop depends on DAC quantization and the relaxation oscillator conversion gain. For ±30% process variations and a 7-bit DAC, the maximum frequency error is ±0.23%. This error is mapped to only 4.6 kHz error in the filter center frequency, which is quite tolerable for Bluetooth application. The 7-bit DAC is implemented using resistive string to insure monotonicity and hence stability of the tuning loop. A non-systematic error should also be considered due to the mismatches between the transconductance and capacitance in the passive RC LPF and the oscillator. These mismatches can add roughly 1% error to the frequency tuning. This is equivalent to another 20 kHz error in the center frequency of the filter, which is still within the tolerable range of a Bluetooth filter. The advantage of such tuning circuit architecture over the conventional PLL-based frequency tuning is that it does not need a low-frequency LP loop filter, which consumes considerable area of the PLL. In addition, it uses a square wave relaxation oscillator, which is easier to build and to guarantee oscillations than the sinusoidal oscillator needed in the conventional PLL.

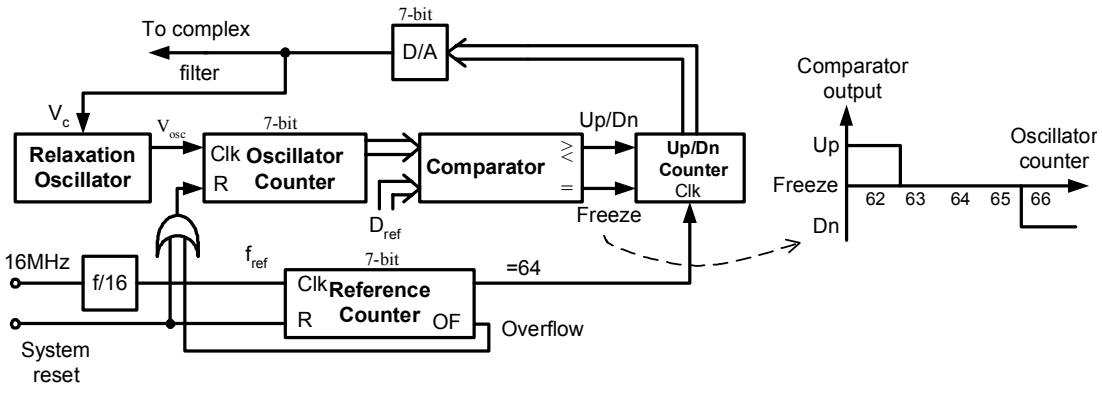


Fig. 3.22 Frequency tuning circuit for complex filter

Guard rings are used to isolate the "noisy" tuning circuit from the filter. An RC LPF is used at the output of the tuning circuit to further attenuate the noise. A 1 MHz tone was observed at the output of the filter at 20 dB below the filter input-referred integrated noise.

3.4.3.2. Relaxation Oscillator

The relaxation oscillator, shown in Fig. 3.23, consists of an OTA, a current switch (M_1-M_6), an integrating capacitor, and a fully differential comparator with hysteresis. The transconductance of the OTA is controlled by changing its common-mode input level (V_C). By applying a constant differential voltage ΔV to the OTA, the output single-ended current will be given by:

$$i_o = g_m \Delta V = K_n' \frac{W}{L}(V_C - V_{Tn})\Delta V \tag{3.12}$$

This output current is mirrored to the tail current source of a differential pair. The output current of the differential pair is integrated on the capacitor C_T. The polarity of that current is controlled by the differential pair transistors. The corresponding slope of the triangular signal is $g_m \Delta V/(4C_T)$. The capacitor voltage is then compared with V_{B1} or V_{B2} depending on the comparator output. The voltages V_{B1}, V_{B2}, and ΔV are obtained from the same resistive string of the DAC used to convert the up-down counter content to analog voltage. The comparator output controls both the differential pair transistors and the threshold voltage of the comparator itself. The oscillation frequency can be expressed as:

$$f_{osc} = \frac{1}{4}\frac{g_m}{C_T}\frac{\Delta V}{\Delta V_B} \tag{3.13}$$

where $\Delta V_B = V_{B1} - V_{B2}$. Since the ratio $\Delta V/\Delta V_B$ is determined by ratio of resistors in the resistive string DAC, it is independent of temperature and process variations and can be predetermined with good accuracy (depending on matching the DAC resistors). Hence, the oscillation frequency is proportional to g_m/C_T with a well-controlled constant of proportionality. The value of C_T is chosen such that, under nominal conditions, the oscillator runs at the reference frequency when the common mode voltage is at nominal (1.65V). The feedback tuning loop ensures that the value of g_m/C remains constant in the presence of temperature and process variations.

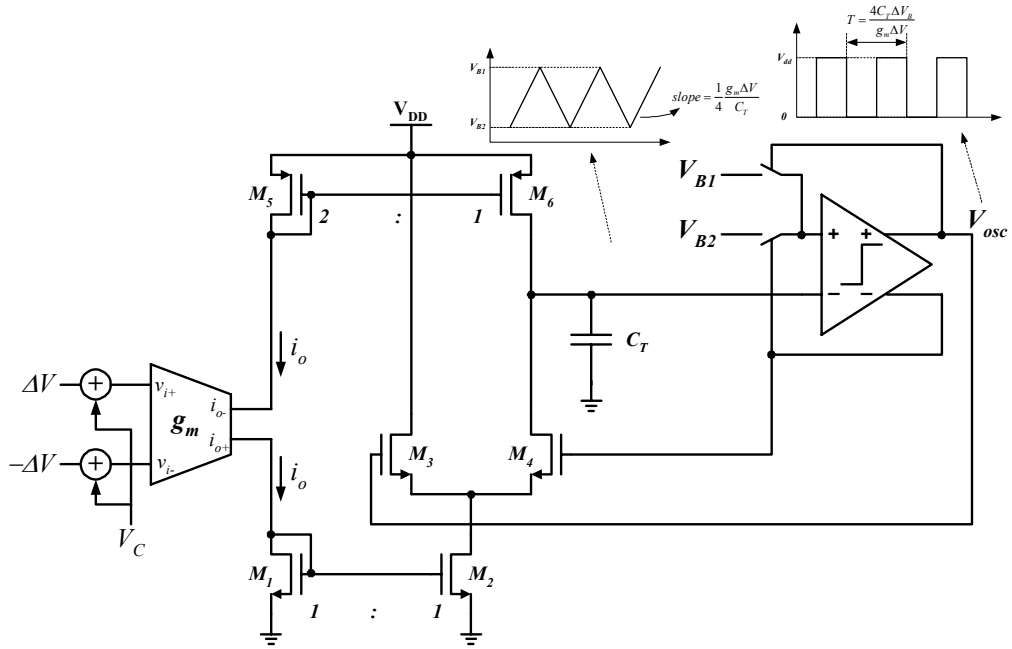


Fig. 3.23 The relaxation oscillator

3.4.4. Experimental Results

The filter and the frequency tuning circuit have been implemented in TSMC 0.35μm CMOS process. The chip micrograph is shown in Fig. 3.24. The areas occupied by the filter and the tuning circuit are $1.6\times0.8\text{mm}^2$ and $1\times0.4\text{mm}^2$, respectively. The filter

operates from 2.7V power supply and draws 4.7mA while the tuning circuit draws 0.8mA. To test the filter frequency characteristics, quadrature sinusoidal signals are used. Polyphase RC network can be used to generate the quadrature signals [18]. However, polyphase RC filters can only generate balanced quadrature signals for a narrow frequency band, and hence, cannot be used to measure the attenuation of interference signals at positive and negative frequencies (relative to the LO frequency). Furthermore, process variations may alter the RC time constant, which will lead to unbalanced quadrature signals. In this case, Tektronix AFG320 signal generator is used to generate the required quadrature signals as shown in the test setup depicted in Fig. 3.25. Fig. 3.26 shows the filter frequency response for the signal and image sides. The figure shows that the image rejection ratio is more than 45dB, which is enough for Bluetooth specifications [1]. The filter attenuates the first and second adjacent channels by 27 and 58dB, respectively. The filter linearity is quantified in terms of spurious-free dynamic range (SFDR). The SFDR is measured by applying two tones at the following frequencies:

$$f_1 = f_C + n \times 1MHz \quad \text{and} \quad f_2 = f_C + n \times 2MHz \tag{3.14}$$

Where n is the two tones separation in MHz, f_C is the filter center frequency, f_1 is the frequency of the first tone, and f_2 is the frequency of the second tone.

Fig. 3.27 shows the measured SFDR versus n. The in-band SFDR (n=0) is about 45.2dB. Fig. 3.28 shows that the in-band two-tone test from which the in-band SFDR is measured. Since the filter is followed by a hard limiter (Fig. 3.7), the in-band SFDR is not a critical parameter in this case. The out-of-band SFDR is a more important parameter to measure. In Bluetooth, the IP3 is calculated for two interferers at 3 and 6MHz away from the desired signal on one side. The IP3 can be approximately

calculated to be about 1.5×SFDR = 91.65dB above the noise floor. The total input referred noise is 29μV_{rms} and the pass band gain is 15dB. Fig. 3.29 shows the measured group delay of the filter, from which it is seen that the in-band group delay variation is about 0.6μs. The asymmetry in the magnitude and group delay responses in Figs. 17 and 20 is a result of parasitic components and mismatches between filter components. Table 3.1 summarizes the experimental results of the filter.

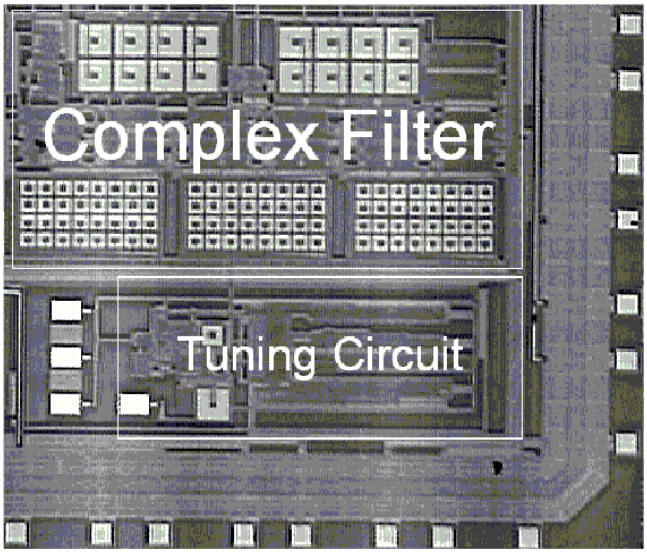


Fig. 3.24 The die photo (filter area = 1.6×0.8mm^2 and tuning circuit area = 1×0.4mm^2)

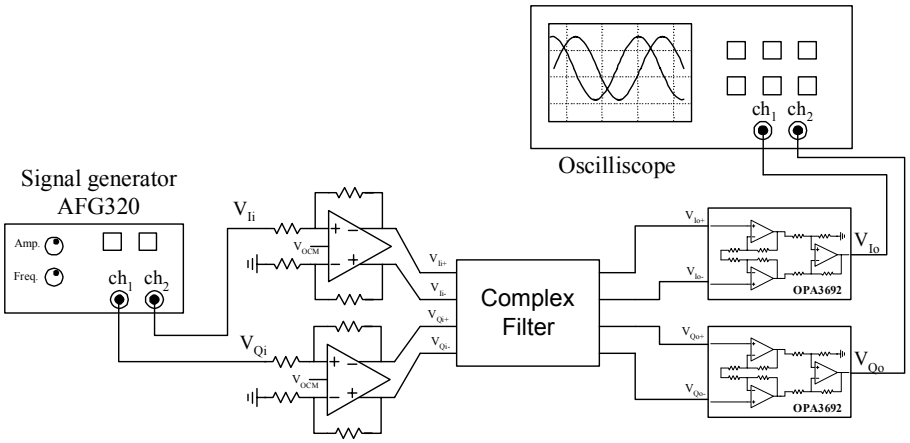


Fig. 3.25 Test setup for the complex filter

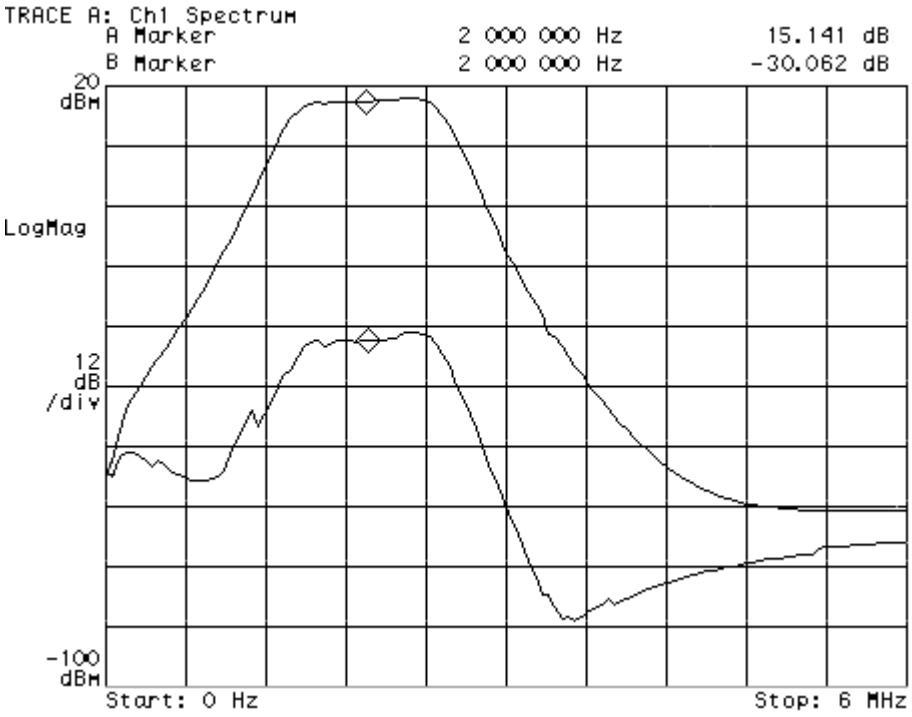


Fig. 3.26 Frequency response at signal and image sides (vertical axis 12dB/div, … ideal, — actual)

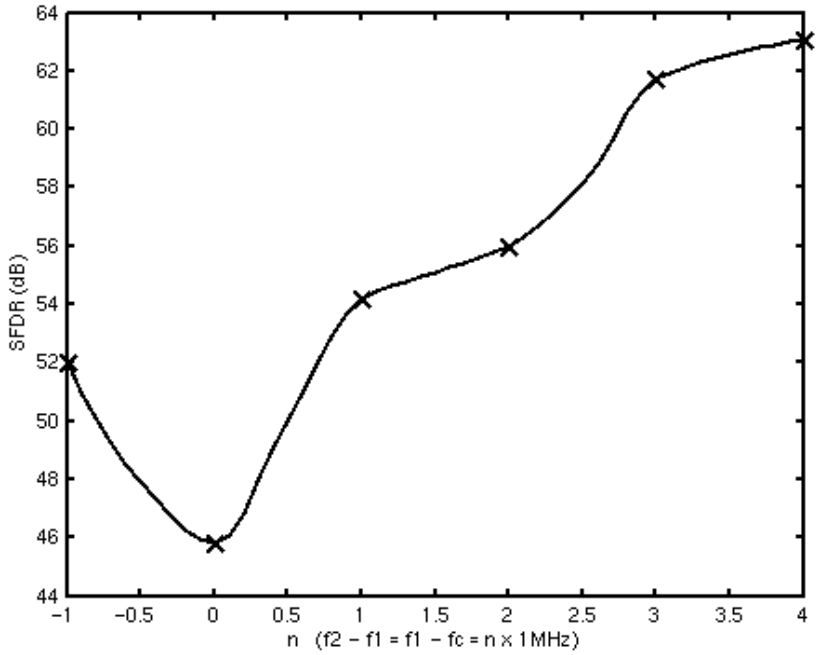


Fig. 3.27 The measured SFDR versus n

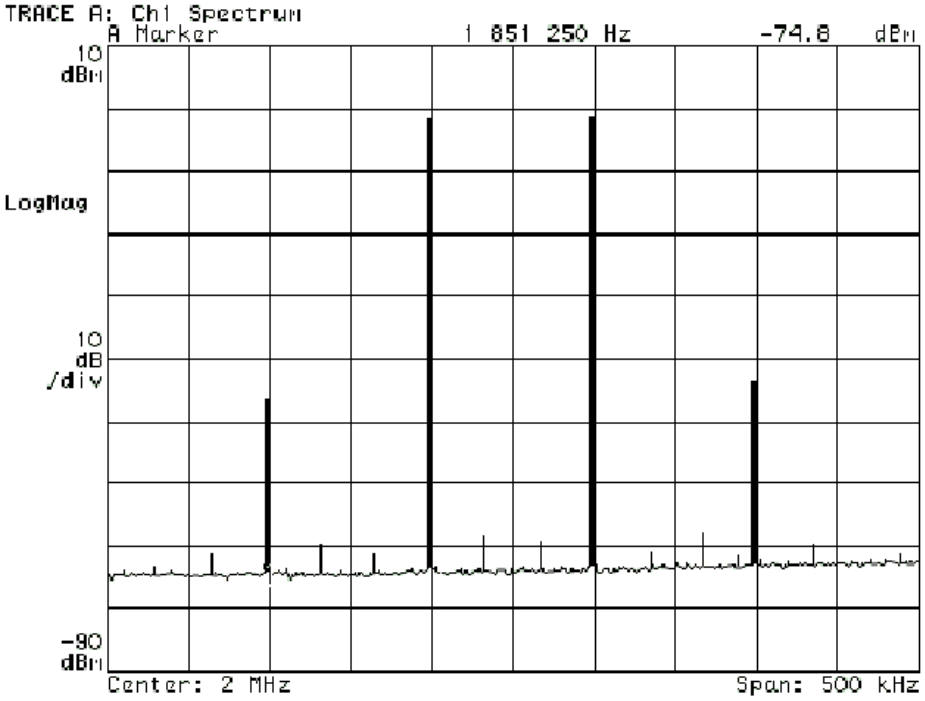


Fig. 3.28 IM3 test for f_1 = 1.95MHz and f_2 = 2.05MHz

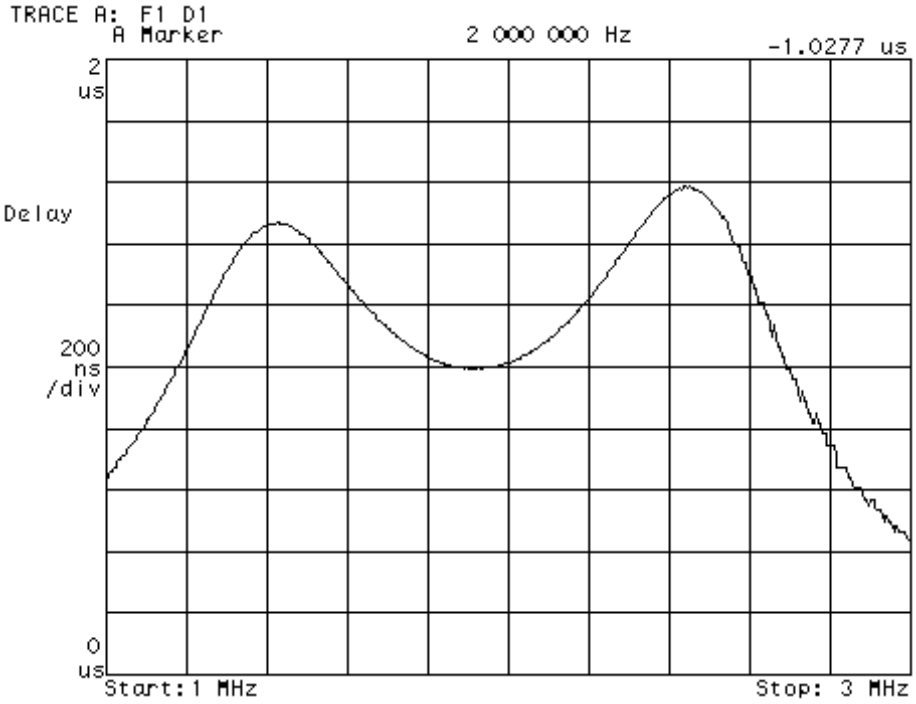


Fig. 3.29 Group delay response

Table 3.1 Summarized filter testing results

Parameter	Value
Center frequency	2MHz
-1dB bandwidth	1.47MHz – 2.53MHz
Pass-band gain	15dB
Input referred noise	$29\mu V_{rms}$
V_{dd}	2.7v
Filter current drain	4.7mA
Image rejection ratio	>45dB
Attenuation @ $f_c \pm 1MHz$	29dBc
Attenuation @ $f_c \pm 2MHz$	58dBc
CMRR	> 50dB
In-band SFDR	45.2dB
SFDR at 3 & 6MHz	61.1dB
In-band group delay variation	0.6μs
Area (filter + tuning circuit)	$(1.28 + 0.4)mm^2$

3.5. Conclusions

A pseudo differential OTA-C complex filter design for low-IF Bluetooth receiver has been presented. The main highlights for the design are: (i) a pseudo differential OTA is used to comply with low voltage operation, (ii) a sound scheme for common-mode control is implemented using a minimum number of CMFF and CMFB circuits. Compared to using a separate CM control circuit for each OTA [20], the proposed CM control scheme roughly consumes only about 1/3 of the area and power of CM control circuitry in conventional schemes, (iii) A non-conventional frequency tuning circuit architecture is used that has advantages over the conventional PLL, in terms of silicon area and design complexity.

CHAPTER IV

DIRECT DIGITAL FREQUENCY SYNTHESIZER (DDFS)

In modern wireless communication systems, fast frequency switching with fine frequency steps is crucial. An example of such a system is Bluetooth, where the signal modulation is GFSK with about 160kHz frequency deviation. A simplified diagram of the Bluetooth transmitter is illustrated in Fig. 4.1, where the input bit stream is pulse shaped by the Gaussian filter and then frequency modulated with frequency deviation (f_d) of 160kHz. The frequency modulator has to provide quadrature outputs that are needed for the quadrature mixer operation.

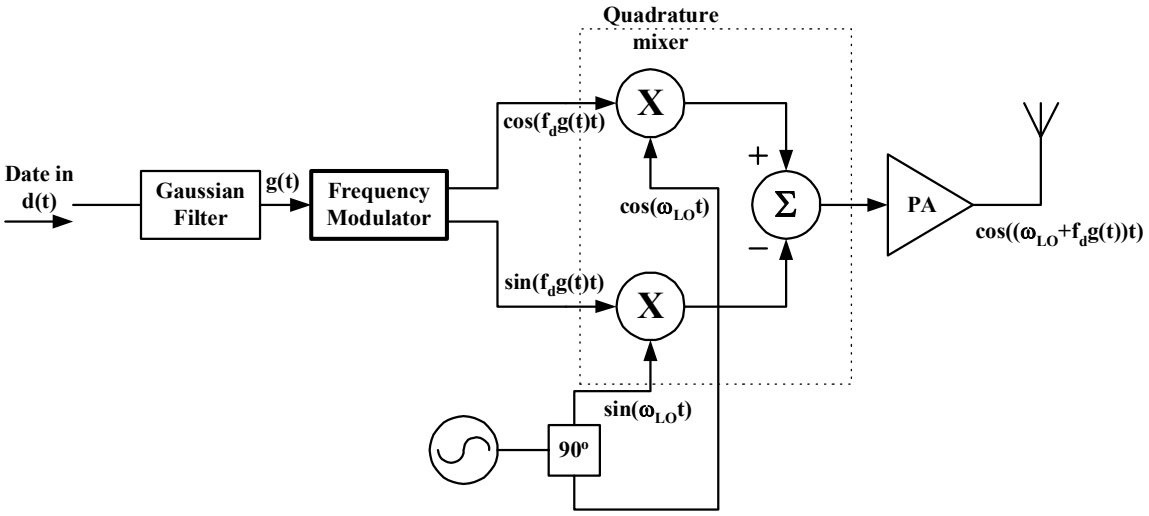


Fig. 4.1 Bluetooth transmitter architecture

There are several possible implementations for the frequency modulator. A traditional PLL based synthesizer is not suitable in these applications due to the inherent loop delay. Another limitation of a PLL is the small range of frequency locking and the

limited frequency resolution. Open loop voltage controlled oscillator (VCO) is also not suitable due to the limited control on the output frequency. A direct Digital Frequency Synthesizer (DDFS) is the most suitable implementation due to its fast frequency switching, with small frequency steps over a wide band. Note also that PLL and VCO implementations are not suitable for the frequency modulator in Fig. 4.1 since the output frequency centered around 0. Since the DDFS implements the sine and cosine functions directly, it can be used in frequency modulator in Fig. 4.1. Finally, the DDFS can generate quadrature output phase very accurately, unlike the PLL where mismatches between components may result in phase mismatch between quadrature outputs. The implementation of the quadrature output frequency modulator is described in detail in the next few sections.

4.1. Conventional DDFS Architecture

Conventional ROM-based direct digital frequency synthesizers (DDFS), as shown in Fig. 4.2, are able to meet the above requirements by storing the values of the sine function in a ROM and scanning these values at a rate proportional to the desired frequency. The digital ROM output is converted to analog using a digital to analog converter (DAC).

The main factors that determine the signal purity in this architecture are: (1) the phase quantization due to finite resolution of the phase accumulator, (2) amplitude quantization noise due to finite resolution of the DAC, and (3) static and dynamic non-idealities of the DAC. The ROM size is exponentially proportional to the desired phase resolution, resulting in huge area consumed by the ROM for reasonable phase resolutions. Moreover, the ROM should be addressed at a much higher rate than the

desired output frequency for moderate spectral purities. Scanning the ROM at high speed makes it power hungry and then, unsuitable for portable wireless applications.

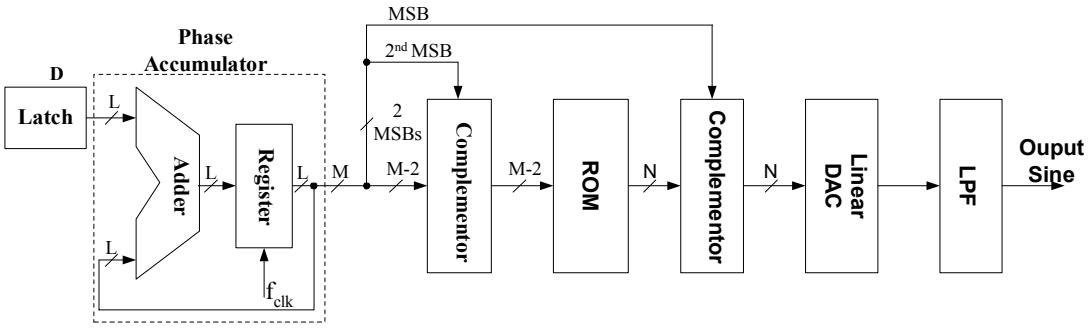


Fig. 4.2 Conventional ROM-based DDFS

Several attempts have been made to reduce the ROM size by using various techniques. The first category of solutions is based on trigonometric identities; the simplest of which is the quarter wave symmetry in the sine function (see Fig. 4.2). Other trigonometric formulas have been used to split a large ROM into two smaller coarse and fine ROMs [27]. The second category of solutions approximates the sine function over the first quarter period by another function, f(x), that can be easily implemented, as illustrated in Fig. 4.3. In this implementation, a ROM look-up table is used to store the error, sin(x)-f(x), which results in less memory word-length requirements. The simplest form is the sine-phase difference method [28] that uses a straight-line approximation for the sine function, i.e., f(x)=x. In this scheme 2 bits of memory word-length are saved. Parabolic approximation has been also introduced [29], which results in saving 4 bits of memory word-length.

However, the above techniques are still consuming considerable area and power, hence not suitable for low cost portable applications. The third category of solutions is to

use a combination of small ROM to store few sample points and a linear interpolation between these points for full computation of the generated sine function as shown in Fig. 4.4. It has been shown that this technique is efficient and the hardware cost required for the additional calculations is shown to be lower than the first two categories [30].

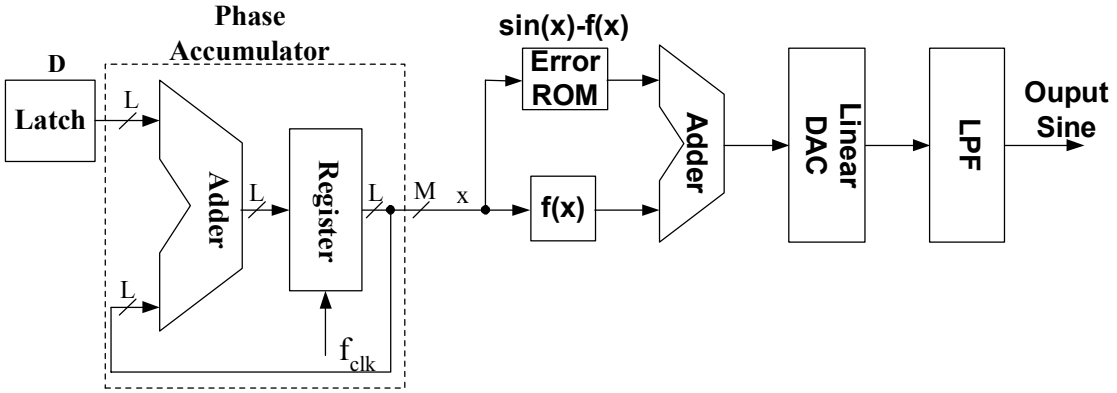


Fig. 4.3 Compression technique using approximation f(x) of the sine function

Another approach was adopted in [31] to avoid using ROM by using nonlinear DAC as shown in Fig 4.5. In this architecture, the non-linear DAC is used to achieve the function of the phase to amplitude conversion and the digital to analog conversion at the same time. This approach was shown to provide considerable area and power savings compared to the conventional approach since the ROM is removed. There were two options for the DAC implementation.

The first implementation is based on resistive string, which consumes less power but is inherently slow (f_{clk}=25MHz) and occupies significant area (1.7mm×1.7mm in 0.5μm technology) due to the large number of resistors and transistors used. The second implementation of the nonlinear DAC is using current-mode techniques to enhance the speed at the expense of power consumption (92mW at f_{clk}=230MHz). To further reduce

the power consumption and die area, a technique was proposed in [32] to split the nonlinear DAC into a coarse DAC and a fine DAC.

In this chapter, a different approach is chosen to implement ROM-less DDFS based on piecewise linear approximation of the sine function as will be presented in the next section [33]. The proposed architecture is shown to have significant area and power savings at high clock rates. Design considerations of the building blocks will be discussed in sections 4.3-4.5. Testing results of the DDFS, as well as a comparison with recently published work, will be presented in section 4.6. Finally concluding remarks are drawn in section 4.7.

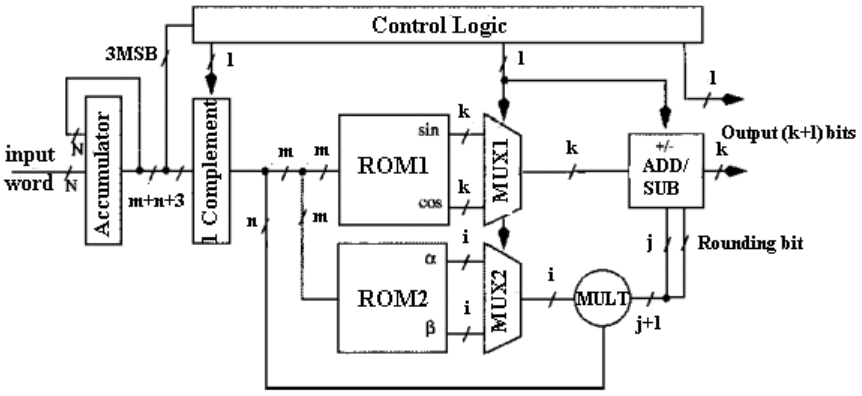


Fig. 4.4 Combining a small ROM and linear interpolation to generate sine function

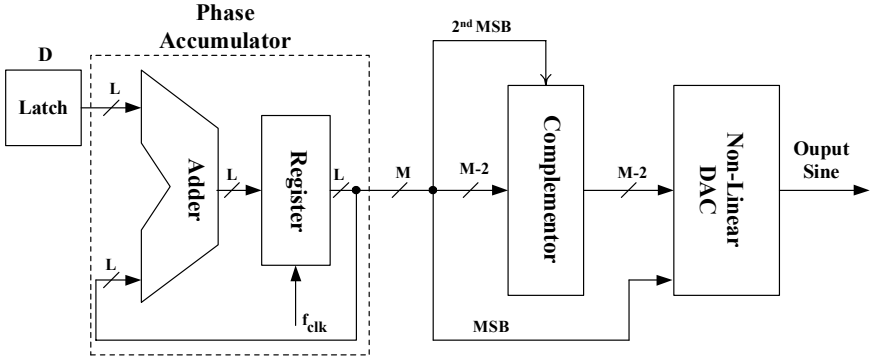


Fig. 4.5 ROM-Less DDFS using non-linear DAC

4.2. Proposed DDFS Architecture

The proposed architecture is based on the idea of breaking the sine function into linear segments as shown in Fig. 4.6, where four segments are shown for the purpose of illustration.

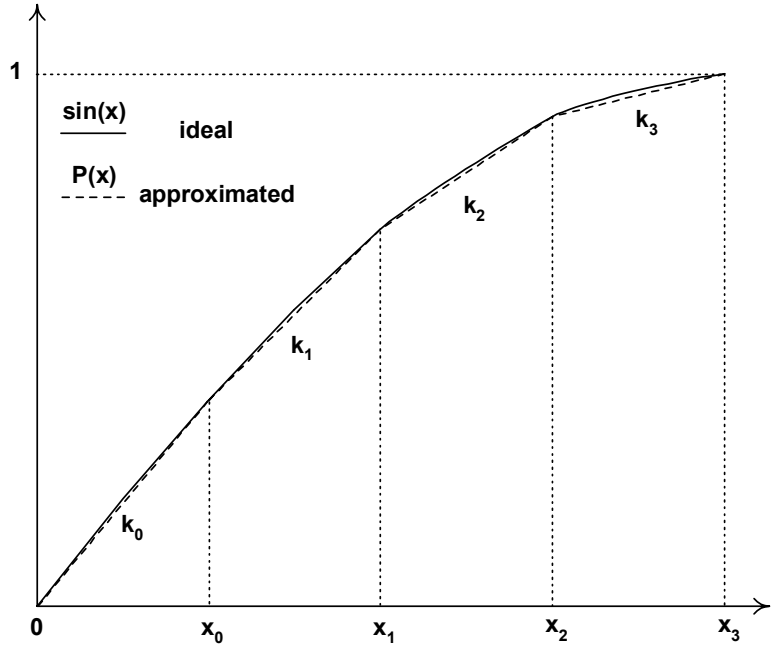


Fig. 4.6 Sine wave approximation

For a given number of segments, the segments' slopes (k_0, k1, ……) are chosen to minimize the integrated mean square error between the ideal, sin(x), and the approximate piece-wise linear, P(x), curves. In order to simplify the implementation of this approximation, the number of segments is chosen to be in powers of 2. The points X_i (i = 0,1, ...) are selected to be equally spaced to further simplify the design. A MATLAB code (Appendix A) is developed to determine the optimum set of slope values, i.e., given the number of piecewise segments, the slope values yielding *minimum mean square error* (MMSE) are determined. Where MMSE is expressed as:

$$MMSE = \min \int_{0}^{x=\frac{\pi}{2}} [\sin(x) - P(x)]^2 dx \tag{4.1}$$

The piecewise linear function is implemented using the block diagram shown in Fig. 4.7. The most significant M bits of the phase accumulator output are used in the phase-to-amplitude (P/A) converter. The P/A converter consists of the complementor, the linear DAC, and the switched weighted-sum (SWS) blocks. The first two MSBs of the accumulator are used to select the quarter in the sine wave cycle. The next M-2 bits are fed into the complementor, whose output is then split into two parts: α and β where α is the MSB part (a bits long), and is corresponding to the segment number, and β is the LSB part (b bits long) which is applied to the input to the DAC. For a given phase resolution (M), the proposed architecture uses an (M-2-a) bits linear DAC. Whereas, in the architecture presented in [31], an (M-2) bits nonlinear DAC is needed. Reducing the DAC bits saves significant area and promotes faster operation. The output of the SWS block is given by:

$$\begin{aligned} V_{sws} &= \sum_{n=0}^{\alpha-1} k_n V_{\min} + k_\alpha V_{DAC} + \sum_{n=\alpha+1}^{2^a-1} k_n V_{\max} && for\ E = 1 \\ V_{sws} &= V_{\min} && for\ E = 0 \end{aligned} \tag{4.2}$$

Where V_{DAC} is the linear DAC output in the range of V_{min} to V_{max} , and E is a digital signal that enables the SWS block depending on which half cycle of the sine function, positive or negative, is currently running. The SWS block in each branch implements half of the sine wave cycle, and hence the differential output voltage is a full sine wave. The frequency of the output sine wave is given by:

$$f_{out} = \frac{D.f_{clk}}{2^L} = D.f_{\min} \tag{4.3}$$

Where D is the digital accumulator input, L is the number of bits in the accumulator, f_{clk} is the input clock frequency, and f_{min} is the minimum synthesized frequency (frequency resolution) that can be obtained.

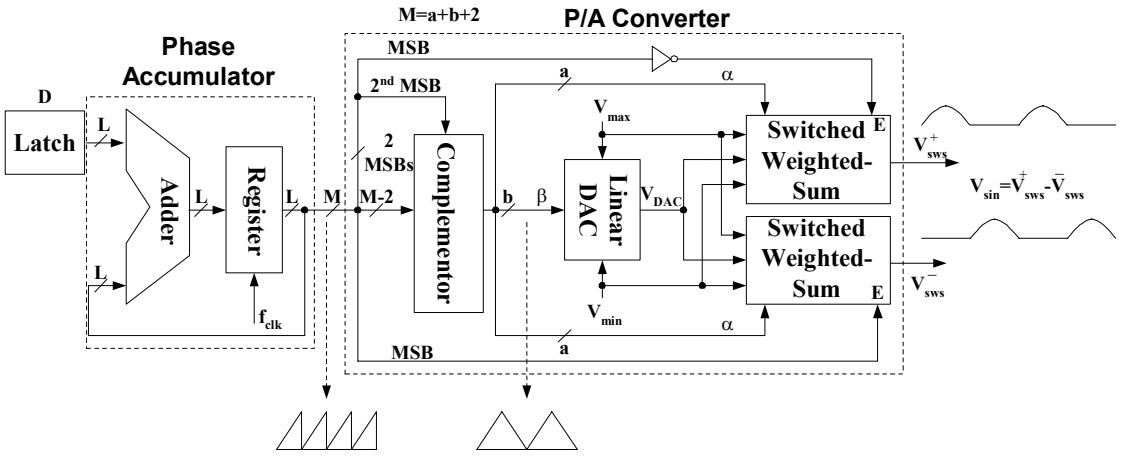


Fig. 4.7 Block diagram of the proposed DDFS architecture

In the proposed design, L=16 and f_{clk}=100MHz yielding a frequency resolution of 1.5kHz. To allow a high clock rate, a 4-stages carry look-ahead adder (4 bits/stage) is used in the accumulator. Sources of distortion in the above architecture are the limited number of segments (2^a), phase resolution (M), and slope resolution. The effect of each of these parameters on the performance is simulated using MATLAB. Fig. 4.8 shows the effect of the number of segments on the spurious free dynamic range (SFDR). Note that the SFDR improves by 12 dB by doubling the number of segments. The SFDR is 59dBc and 71dBc for 8 (a=3) and 16 (a=4) segments, respectively.

In Bluetooth, the transmitter spurious emissions should be less than –20dBc to -40dBc at 2MHz and -40dBc to -60dBc at 3MHz, depending on transmitter power class. Since the 16-segments DDFS consumes about 50% more power and area than the 8-segments design, as will be shown at the end of section 4, 8-segments design is adopted.

A passive RC pole can be used at the DDFS output to attenuate the spurs at frequencies far from the fundamental output. For 8 segments, the effect of the finite phase resolution is shown in Fig. 4.9. It shows that no significant improvement in the SFDR is achieved for phase resolutions more than 10 bits. For this choice of M=10 and a=3, only 5-bit DAC (b=M-2-a=5) is needed.

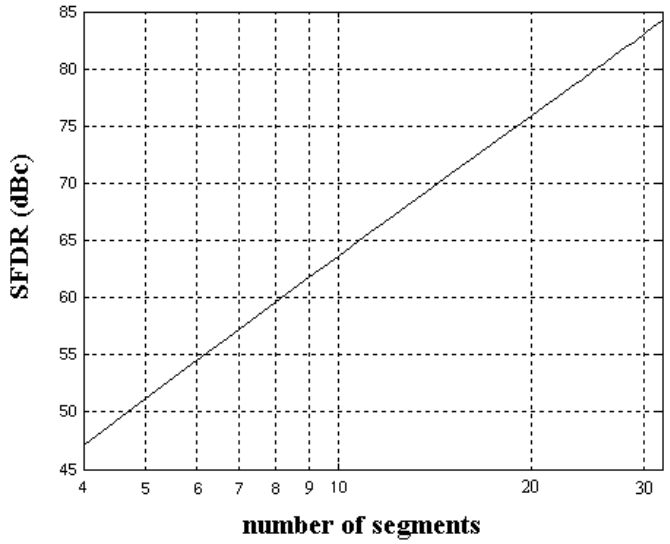


Fig. 4.8 Effect of the number of segments on the SFDR

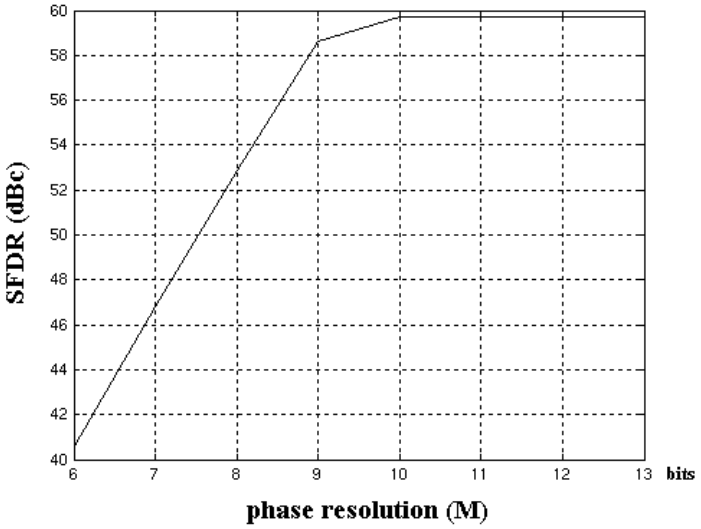


Fig. 4.9 Effect of phase resolution on the SFDR

The weighted-sum function is implemented using resistive dividers (as will be discussed later). Each resistance is an integer multiple of a unit resistance ΔR, which determines the slope resolution. The effect of finite unit resistance is shown in Fig. 4.10 for 8 segments and 10 bits of phase resolution.

Note that for a normalized unit resistance ($\Delta R/R_{max}$) less than 0.4%, there is no much improvement in SFDR. To allow for some margin, ($\Delta R/R_{max}$)=0.2% is a good choice for an 8 segment sine shape. An extra care must be taken in the layout of these resistors to achieve the required resolution.

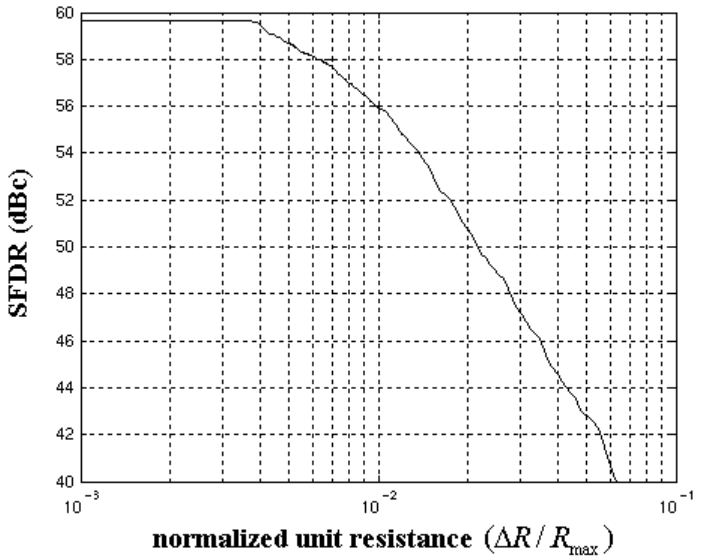


Fig. 4.10 Effect of finite unit resistance on the SFDR

Table 4.1 summarizes the required values of phase resolution (M), normalized unit resistance, and the corresponding SFDR for different number of segments. Note that doubling the number of segments, i.e., increasing 'a' by 1 requires 2-bit increase in the phase resolution. This implies doubling the linear DAC size, i.e., increasing 'b' by 1.

Quadrature outputs are generated by replicating the P/A converter. The 90^0 phase shift is implemented by adding 01 to the two MSBs of the accumulator output.

Table 4.1 Required values of (M), ($\Delta R/R_{max}$), and corresponding SFDR for different number of segments

Number of segments	Phase resolution (M)	($\Delta R/R_{max}$)	SFDR
4	8 bits	0.7 %	47 dBc
8	**10 bits**	**0.4 %**	**59 dBc**
16	12 bits	0.2 %	71 dBc
32	14 bits	0.1 %	83 dBc

4.3. Linear DAC

The 5-bit resistive string DAC is implemented as shown in Fig. 4.11. In order to have smoother transitions at the corner points between segments in the output sine wave, a ½ LSB offset is introduced to the DAC output by using R/2 value for the lower and uppermost resistors. A 5-to-32 decoder is used to turn ON the switch corresponding to the digital input β. At any given time, only one switch must be turned ON. Due to non-equal delays at the decoder outputs, more than one switch can be turned ON at the same time or all the switches can be turned OFF. This will result in undesirable glitches at the output of the DAC. To solve this problem, the outputs of the decoder are sampled at $\overline{CLK}$.

The settling time for each digital input word is determined by three factors: (1) The total capacitance (C_L) at the output node V_{DAC} which is dominated by the drain capacitances of all the switches and the input capacitance of the next stage. (2) The input resistance of the resistive string seen from the corresponding node (m). The closer the node to the middle of the resistor string, the higher the resistance seen. The value of R is

chosen as a trade-off between power consumption and settling time. For this particular design, R=50Ω is found to be a good compromise. (3) The ON resistance, R_{ON}, of the corresponding switch M_m. In order to minimize the load capacitance C_L, different transistors' widths are used to have the same settling time for all digital input combinations. To allow for large overdrive voltage in the NMOS switches, the terminal voltages (V_{min} and V_{max}) of the resistor string are chosen to be 0V and 0.5V, respectively.

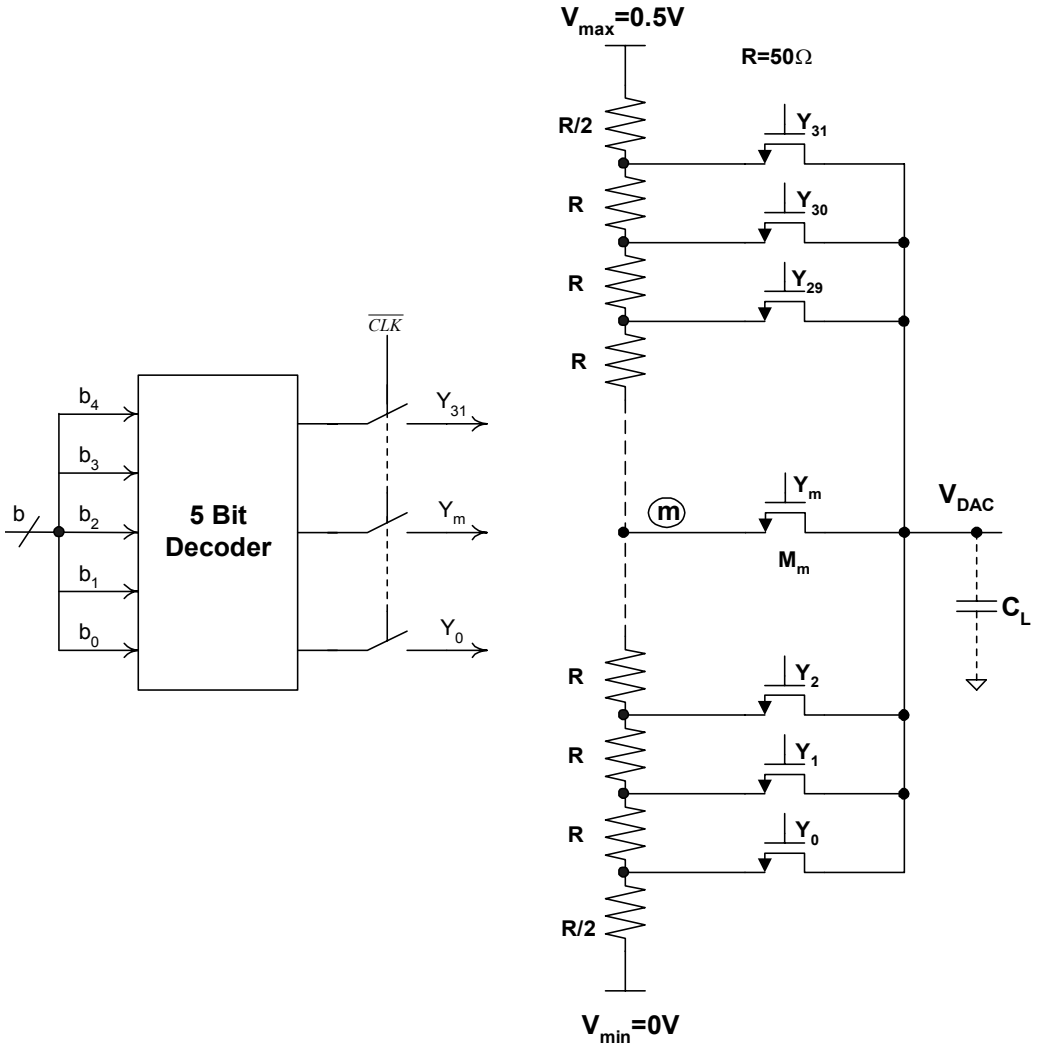


Fig. 4.11 Linear DAC circuit implementation

4.4. Switched Weighted-Sum Block

Two SWS blocks are used in the system, one for each half of the sine wave cycle. The details of the SWS block are shown in Fig. 4.12.

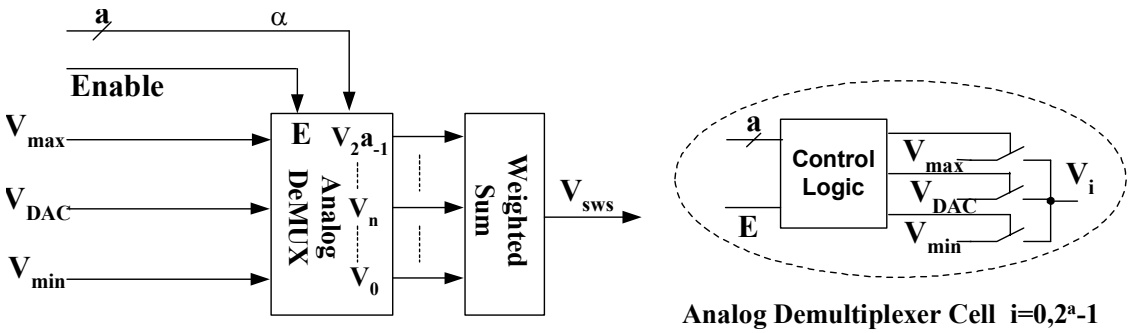


Fig. 4.12 Switched weighted-sum block

The output of the SWS block can be written as:

$$V_{sws} = \sum_{n=0}^{2^a-1} k_n V_n \tag{4.4}$$

The role of the analog demultiplexer is to route V_{max}, V_{DAC}, or V_{min} to V_n based on the value of α, corresponding to the segment number, and enable E as follows:

$$\begin{aligned} V_n &= V_{\max} \quad for\ \ n < \alpha \ \ \&\ E = 1 \\ V_n &= V_{DAC} \quad for\ \ n = \alpha \ \ \&\ E = 1 \\ V_n &= V_{\min} \quad for\ (n > \alpha \ \ \&\ E = 1)\ \text{OR} \ \ E = 0 \end{aligned} \tag{4.5}$$

The analog demultiplexer consists of 2^a cells, one for each output. The basic analog demultiplexer cell for each output V_i for i=0, 2^a-1 is illustrated in the same figure.

The weighted-sum function described in equation (4.4) is implemented using resistors and buffers as shown in Fig. 4.13. Assuming ideal buffers (unity gain and zero output resistance), the weighted-sum output V_{sws} is given by:

$$V_{sws} = \frac{\sum_{n=0}^{2^a-1}(V_n / R_n)}{\sum_{n=0}^{2^a-1}(1/R_n)} = \frac{\sum_{n=0}^{2^a-1}(V_n / R_n)}{G_T} \tag{4.6}$$

Where $R_T=(1/G_T)$ is the parallel combination of all resistors in the weighted-sum network. Comparing equations (4.4) and (4.6), we get:

$$k_n = \frac{R_n}{R_T} \qquad n = 0,1,\cdots,2^{a-1} \tag{4.7}$$

Equation (4.7) is used to obtain the values of R_n from the segments' slopes. Note from equation (4.6) that V_{sws} depends on the ratio of resistors, hence the matching of the resistors in the weighted-sum block in each branch is a critical issue. On the other hand, matching between the resistors of the weighted-sum blocks in both positive and negative branches is not critical.

The buffers are the most power consuming parts in the proposed system. Care must be taken in the design of these buffers to minimize the overall power consumption while keeping good linearity.

Fig. 4.13 shows also the transistor level design of each buffer. PMOS input transistors are used to eliminate the body effect by shorting the source and body terminals (nwell technology is used). Another advantage of using PMOS input transistor is that the input voltage can be as low as 0V. Since this voltage is routed from the DAC resistor string to the buffer input using MOS switches (in the DAC and the analog demultiplexer), then high-speed NMOS switches can be used with high overdrive voltage. This helps to minimize the delay associated with these switches without having to use wide transistors. Note that the output voltage of each buffer is DC shifted by V_{SG} relative to its input. Since all the buffers are biased using the same biasing voltages V_{B1} and V_{B2}, they all

have the same nominal DC shift. However, due to mismatches between transistors in different buffers, this DC shift may differ from one buffer to the other. These DC shift mismatches will result in a slight offset in the output sine wave, but will not affect its spectral purity.

Since buffers have different load resistances, the actual relative slope values of the segmented sine wave will be slightly different from equation (4.7) if the buffers were identical due to the finite buffer output resistance. One way to dilute this effect is to reduce the buffers output resistance by increasing the bias current or increasing W/L of M_1 (Fig. 4.13), which will increase the power consumption or increase the load capacitance to the analog demultiplexer, respectively.

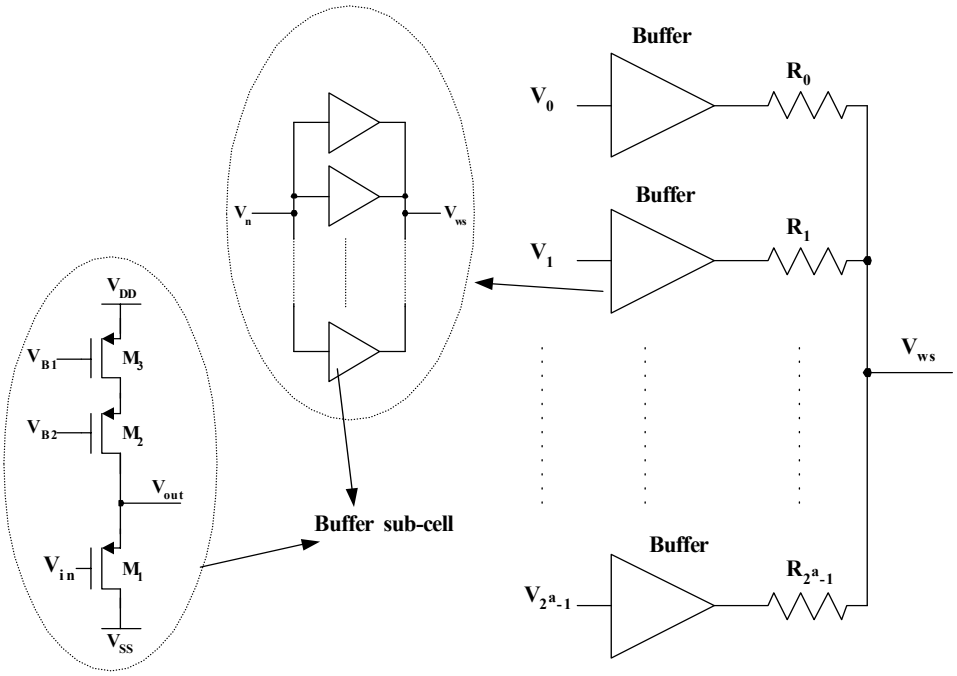


Fig. 4.13 Implementation of the weighted-sum block

Instead, to have the same loading effect on all buffers, each buffer is designed such that the ratio of its output resistance to the load resistance is the same for all buffers. To

allow better matching between buffers, each buffer is designed as a number of parallel buffer sub-cells. These sub-cells are identical in all buffers, but the number of parallel sub-cells in each buffer is inversely proportional to the desired output resistance. As a result, the buffers' output resistances will account for a slight attenuation, but the relative slope values of the segments remain unchanged. Consequently, the total current drain in all buffers will be inversely proportional to the parallel combination, R_T, of all resistors in the weighted-sum network. Hence, scaling of all resistors is very important in determining the overall power consumption. However, the higher the value of R_T, the slower the response of the circuit due to the large time constant R_TC, where C is the input capacitance of the following stage. The settling error within the clock period is given by:

$$settling\ error = \Delta V_{out} e^{-\frac{1}{f_{clk} R_T C}} \tag{4.8}$$

The largest error corresponds to the largest output voltage step $\Delta V_{\max} = 0.5V$. This largest error should be set to be smaller than the smallest output step, which corresponds to the smallest slope. For the particular case when a=3, the smallest slope value (obtained from the MMSE algorithm implemented in MATLAB) is 0.096 for a sine wave of unit amplitude. Therefore, the smallest output step is:

$$\text{smallest output step} = \Delta V_{\min} = \text{DAC step} \times \text{smallest slope} = \frac{0.5}{2^5} \times 0.096 = 1.5mV \tag{4.9}$$

By fixing the settling error in equation (4.8) smaller than the above value, we get:

$$R_T < \frac{1}{C.f_{clk} \ln\left(\frac{\Delta V_{\max}}{\Delta V_{\min}}\right)} = \frac{1}{5.81C.f_{clk}} \tag{4.10}$$

The value of C determines the integrated output noise due to resistors, known as KT/C noise. This noise should be sufficiently smaller than the largest output spur, which

in this case is 59dB below the fundamental tone. If we set the noise level to be lower than this spur, then:

$$\sqrt{\frac{KT}{C}} < \frac{A}{\sqrt{2}\ 10^{(SFDR/20)}} \tag{4.11}$$

Where, K is Boltzmann constant, T is the temperature (assumed 300°K), and A is the output amplitude (0.5V).

In case of 8 segments approximation, the above equation yields C > 0.026pF. This is rather a loose condition for this capacitance, which is typically larger than 0.1pF. In this design, 0.3pF is assumed for the load capacitance, for which the maximum parallel resistance R_T is 4.7kΩ, as given by equation (4.10) for f_{clk}=100MHz. To allow for some margin for process variations, all resistances are scaled to have a nominal parallel combination of 3kΩ. The lowest and highest resistors in the weighted sum network are 15kΩ and 150kΩ, respectively.

An alternative switched capacitor implementation of the weighted-sum block is shown in Fig. 4.14. The cycle period of the two phases φ_1 and φ_2 is $(1/f_{clk})$. Since the phases φ_1 and φ_2 are non-overlapping phases, the on-time period should be chosen less than $(0.5/f_{clk})$. The weighted-sum output V_{sws} is given by:

$$V_{sws} = \frac{-1}{C_F}\sum_{n=0}^{2^a-1} C_n V_n \tag{4.12}$$

Equation (4.12) is valid for phase φ_1, while the output is hold by capacitor C_H during φ_2. Comparing equations (4.4) and (4.12), we get:

$$k_n = \frac{C_n}{C_F} \qquad n = 0,1,\cdots,2^{a-1} \tag{4.13}$$

Equation (4.13) is used to obtain the values of C_n from the segments' slopes. Note from equation (4.13) that V_{sws} depends on the ratio of capacitors, hence the matching of the capacitors in the weighted-sum block in each branch is a critical issue. On the other hand, matching between the resistors of the weighted-sum blocks in both positive and negative branches is not critical.

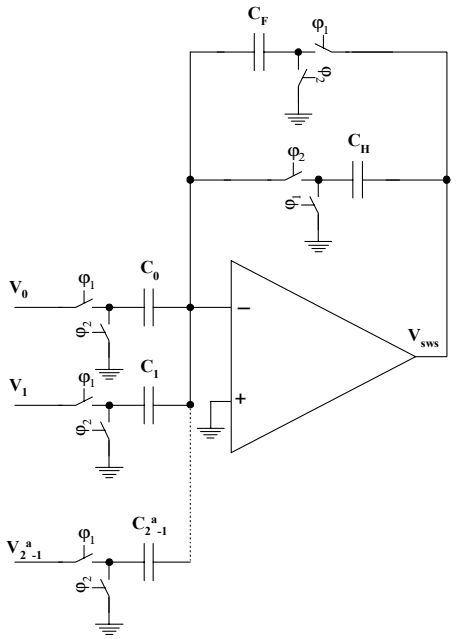


Fig. 4.14 Alternative implementation of the weighted-sum block

Example: The minimum value of the capacitor C_{min} is determined by the noise requirement, which should be sufficiently smaller than the largest output spur. For the case of 8 segments the integrated noise should be 59dB below the fundamental tone. According to equation (4.11) this results in a C_{min} of 0.3pF. Because of the finite GBW of the OpAmp, the output voltage will not settle to its final value during φ_1. To make this error negligible, it is required, as a rule of thumb, that:

$$m \cdot GBW \cdot T > 5 \tag{4.14}$$

Where m is the capacitor ratio between the sum of all the feedback capacitors divided by the sum of all the capacitors connected to the input terminal of the Op Amp, and T is the period of the clock frequency. In our case, m=1/5 according to equation (4.13). Thus, from equation (4.14) and for f_{clk}=100MHz, GBW should be larger than 2.5GHz.

$$GBW > 2.5\times10^9 \Rightarrow \frac{g_m}{2\pi C_{\min}} > 2.5\times10^9 \Rightarrow \frac{I_{tail}}{2\pi V_{ov} C_{\min}} > 2.5GHz \qquad (4.15)$$

A value of V_{ov}=0.5V and C_{min}=0.3pF results in a tail current of the input stage of the OpAmp I_{tail} of 2.4mA. Due to the finite gain of the OpAmp, the steady state settling value deviates from its ideal value. To minimize this error, the gain of the OpAmp should be at least 60dB, which might require a two-stage OpAmp. This makes the design more difficult in terms of stability requirements, i.e., phase margin. For the OpAmp to act as a single pole system with enough phase margin, the frequency of the non-dominant pole ω_{nd} must be pushed at least twice that of GBW, i.e., ω_{nd} > 5Grad/sec. In the used technology (0.5μm process), this means even pushing more current than calculated previously, i.e., I_{tail}. The current consumption of the first solution of the weighted sum block, shown in Fig. 4.13, is 1.5mA as shown in section 5. Since the current consumption of the second solution, shown in Fig. 4.14, is more than 2.4mA, the first solution has been adopted in the final design.

Design procedure: A simple design strategy for the proposed DDFS architecture, for a given SFDR, can be itemized as:

(1) Choose the number of segments of the piecewise linearly approximated sine function according to Fig. 4.8 for the required SFDR.

(2) The values of the slopes of the piecewise linearly approximated sine function are calculated to minimize MMSE as given by equation (4.1)

(3) The required values of phase resolution (M) and resistance mismatch ($\Delta R/R_{max}$) are calculated from Table 4.1.

(4) Calculate the value of C such that the integrated output noise, i.e. KT/C noise, is sufficiently smaller than the largest output spur as given by equation (4.11).

(5) Calculate the value of the maximum parallel resistance R_T such that the settling error is smaller than the minimum output step, ΔV_{min}, for the used clock frequency f_{clk}.

Example: Consider the case of 16 segments (a = 4). The targeted SFDR is 71dB and the smallest slope is 0.05. According to Table 4.1, the minimum phase resolution for 71dB SFDR is 12bits. Hence, the number of DAC bits is $b = M - 2 - a = 6$. From equation (4.11), we get C > 0.42pF. If we take C = 0.5pF (this may require adding a physical capacitance), then the parallel combined resistance will be $R_T < 2.8k\Omega$. If we choose R_T = 2kΩ, this design will consume about 50% more power compared to the 8-segments design. The buffers' area will roughly increase by the same percent.

4.5. Experimental Results

The proposed quadrature (I & Q) output DDFS has been implemented through MOSIS, in 0.5μm AMI CMOS process. The die photo is shown in Fig. 4.15. The chip active area is $1.4mm^2$, of which 25% is occupied by the phase accumulator. On-chip buffers are included to drive the pin capacitance. The DDFS operates from a single 2.7V

supply with 3mA current drain and a clock frequency of 100MHz. The testing setup incorporates an off-chip instrumentation amplifier, as shown in Fig. 4.16, for differential to single-ended conversion. It uses low distortion (THD=-75dB at 1MHz) buffer amplifiers with 1pF input capacitance.

Fig. 4.17 shows the single-ended outputs ($V_{sws}^{+}$and V_{sws}^{-}) of the I branch as well as the differential output when f_{out}=98kHz. Fig. 4.18 shows the two quadrature outputs of the I & Q branches at the same output frequency. The peak-to-peak magnitude is about 910mV, which is slightly less than the ideal $1V_{p-p}$ magnitude due to the expected attenuation of the weighted-sum and the output buffers.

The modulation capabilities of the proposed DDFS have been also tested. Fig. 4.19 shows an example of frequency modulation where the modulating signal is a square wave of frequency 1kHz. An example of amplitude modulation is shown in Fig. 4.20. The modulating signal is applied at V_{max} (see Fig. 4.10). It is a sine wave of frequency 1kHz and peak-to-peak magnitude of 400mV with a DC offset of 250mV. The output spectrum for $f_{out}=(f_{clk}/1024)$ = 98kHz and $f_{out} = (f_{clk}/64)$ = 1.56MHz, at which the SFDR is 57.3dBc and 42.1dBc, are shown in Fig. 4.21 and Fig. 4.22 respectively. The SFDR versus the output synthesized frequency is plotted in Fig. 4.23. The SFDR is better than 59dBc for low synthesized frequencies. For high synthesized frequencies, the SFDR is degraded due to large output steps of the DAC and the switched weighted-sum blocks. Fig. 4.24 shows the SFDR versus the clock frequency for $f_{out}=(f_{clk}/128)$.

The DDFS is shown to operate from a clock frequency up to 130MHz. Higher clock frequencies could not be achieved due to the frequency limitations of the phase accumulator.

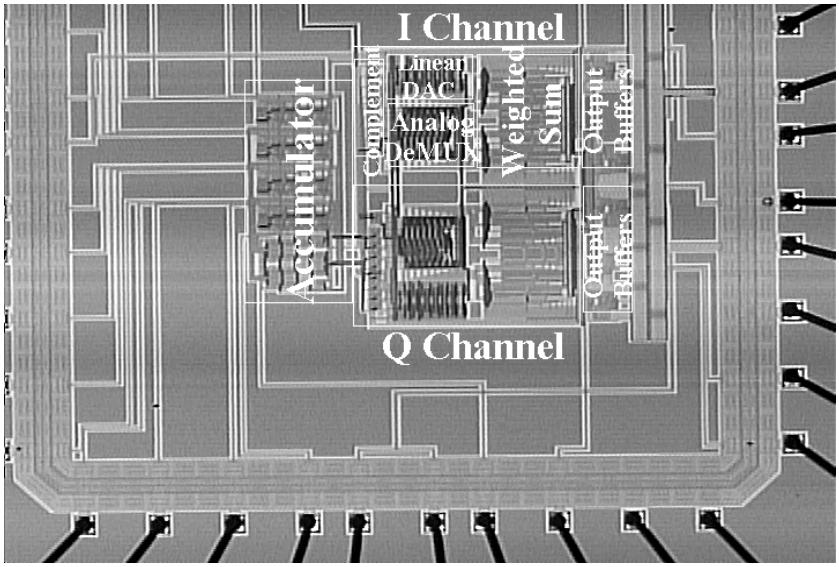


Fig. 4.15 Chip micrograph

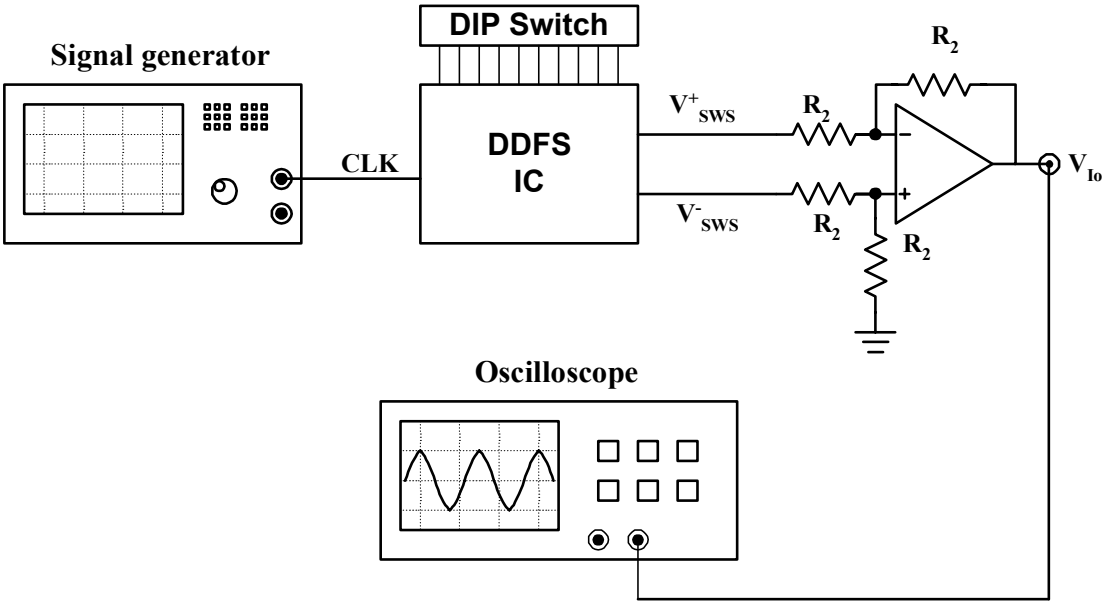


Fig. 4.16 Testing setup

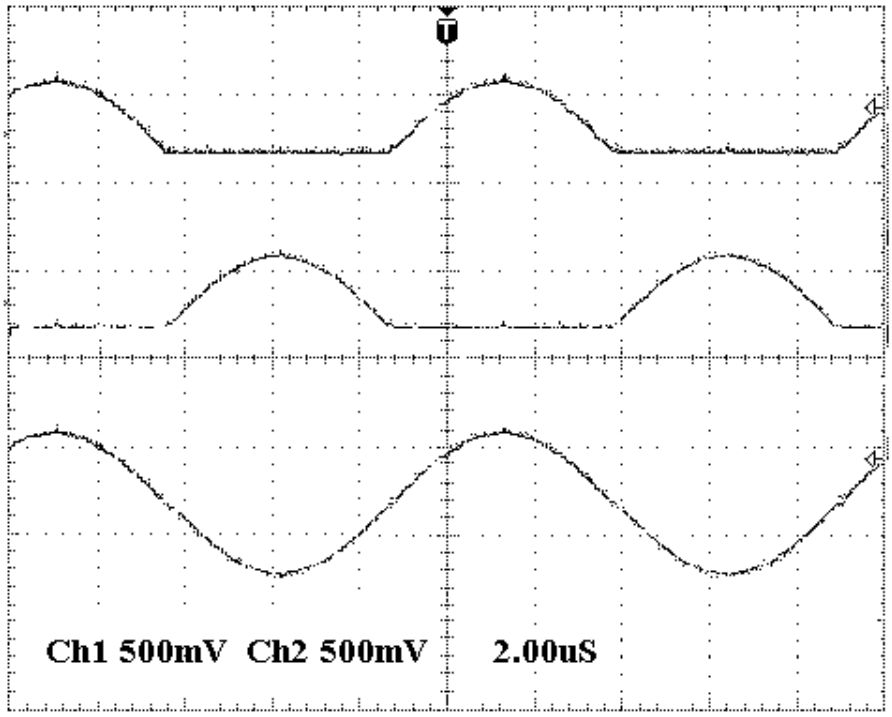


Fig. 4.17 Single-ended and differential outputs of the I branch (f_{out}=98kHz)

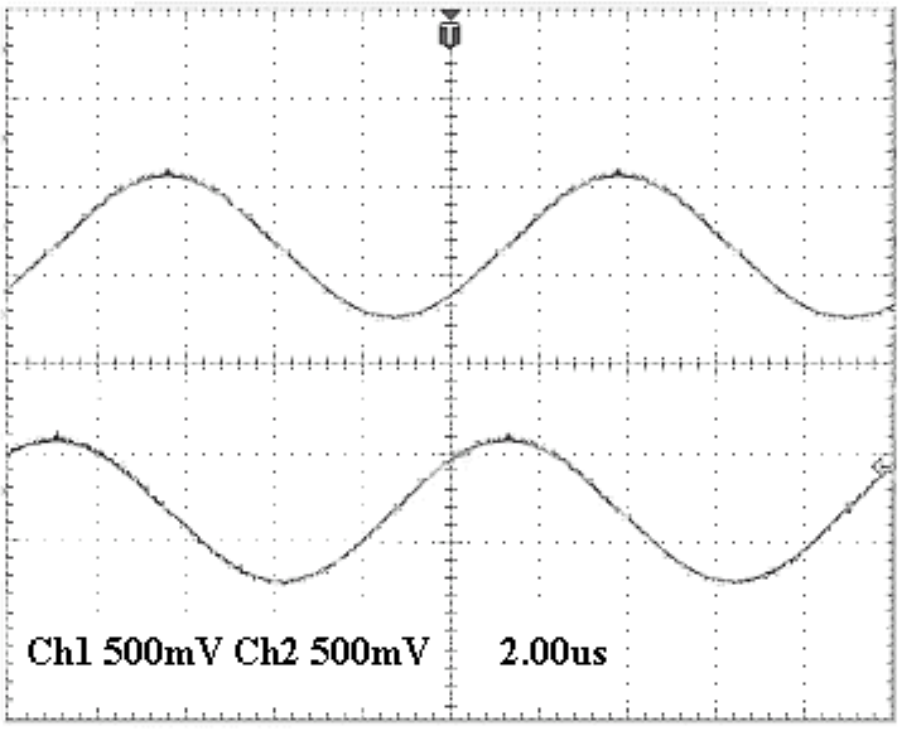


Fig. 4.18 Quadrature outputs I and Q at f_{out}=98kHz

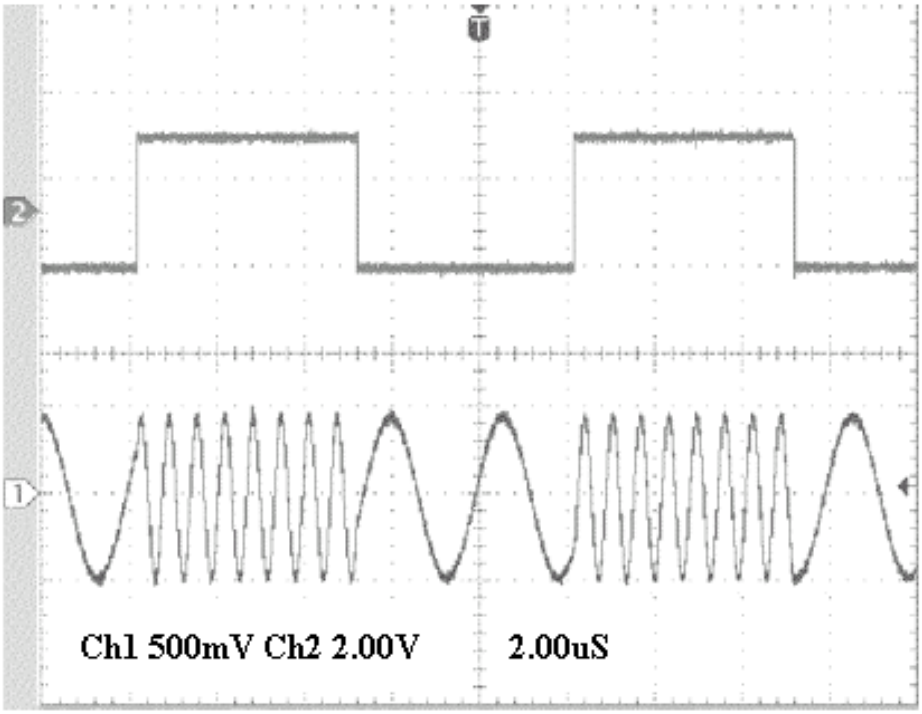


Fig. 4.19 Frequency modulation

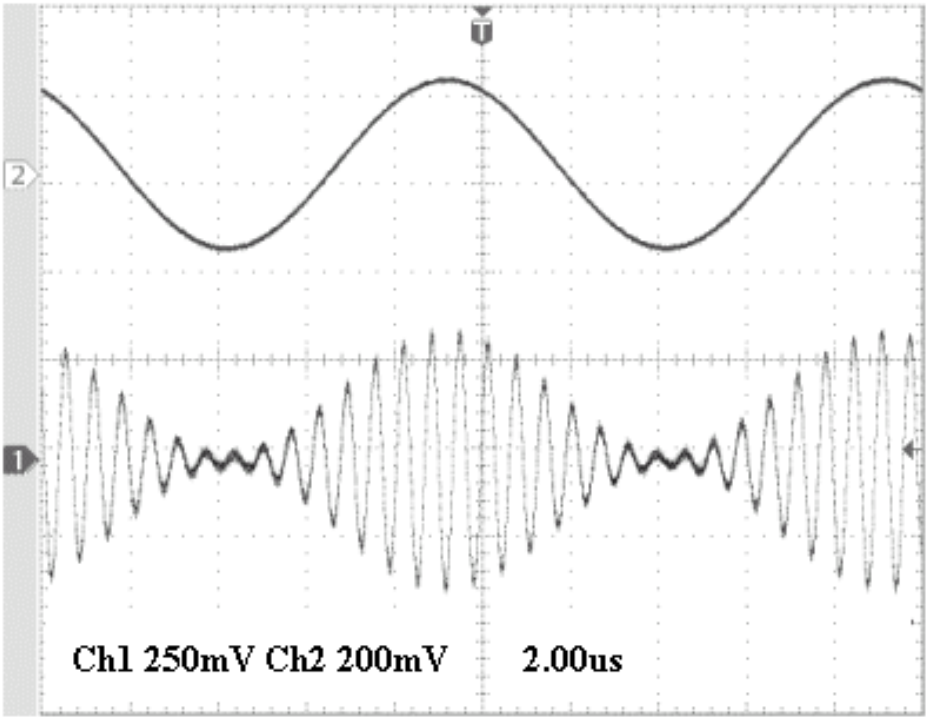


Fig. 4.20 Amplitude modulation

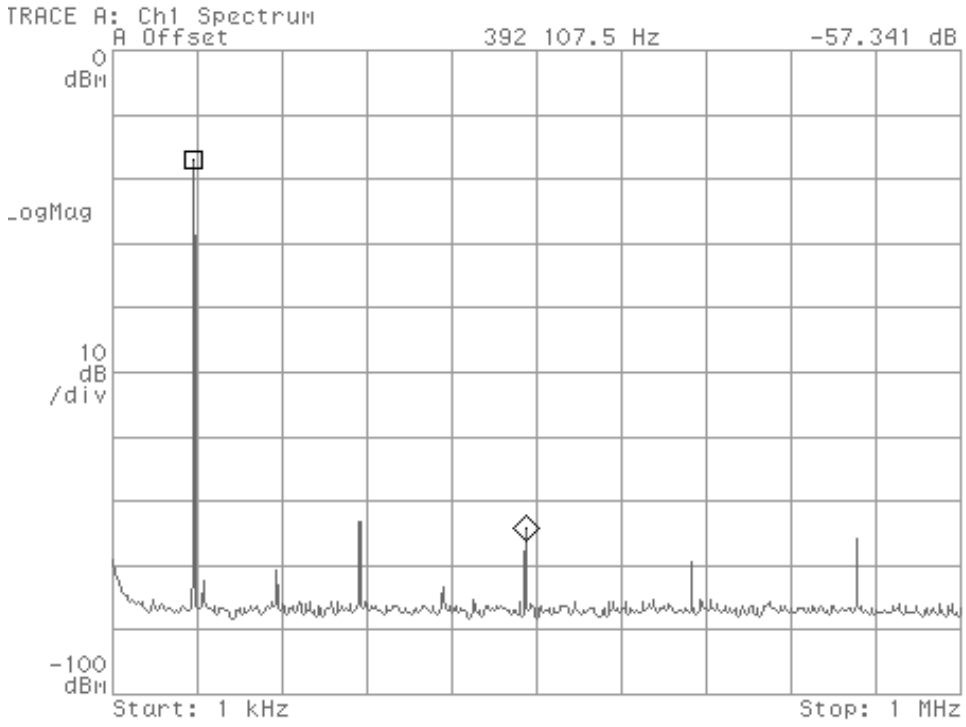


Fig. 4.21 Output spectrum at f_{clk}=100MHz and f_{out}=98kHz with SFDR=57.3dBc

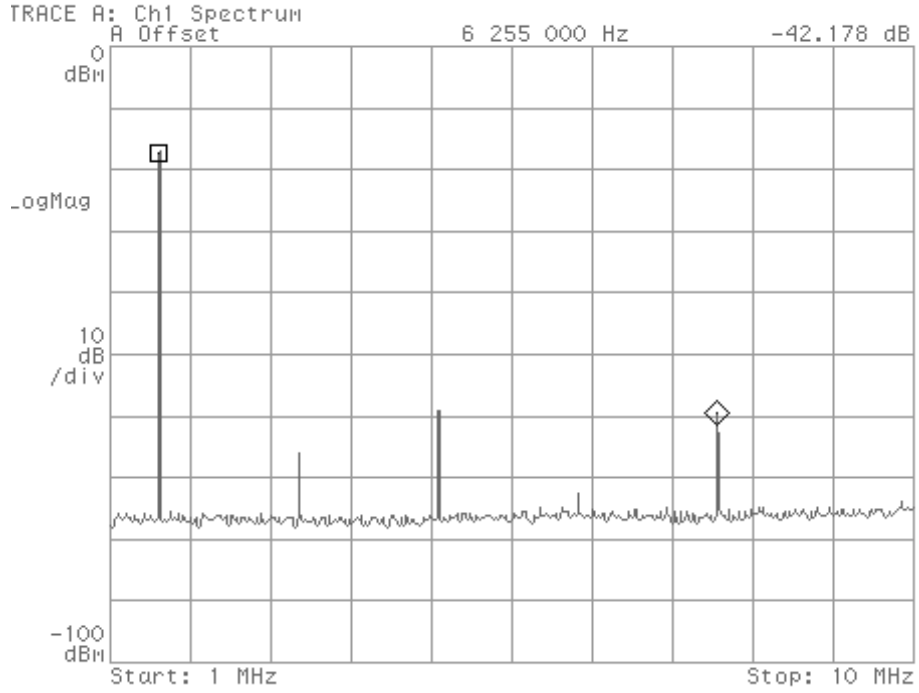


Fig. 4.22 Output spectrum at f_{clk}=100MHz and f_{out}=1.56MHz with SFDR=42.1dBc

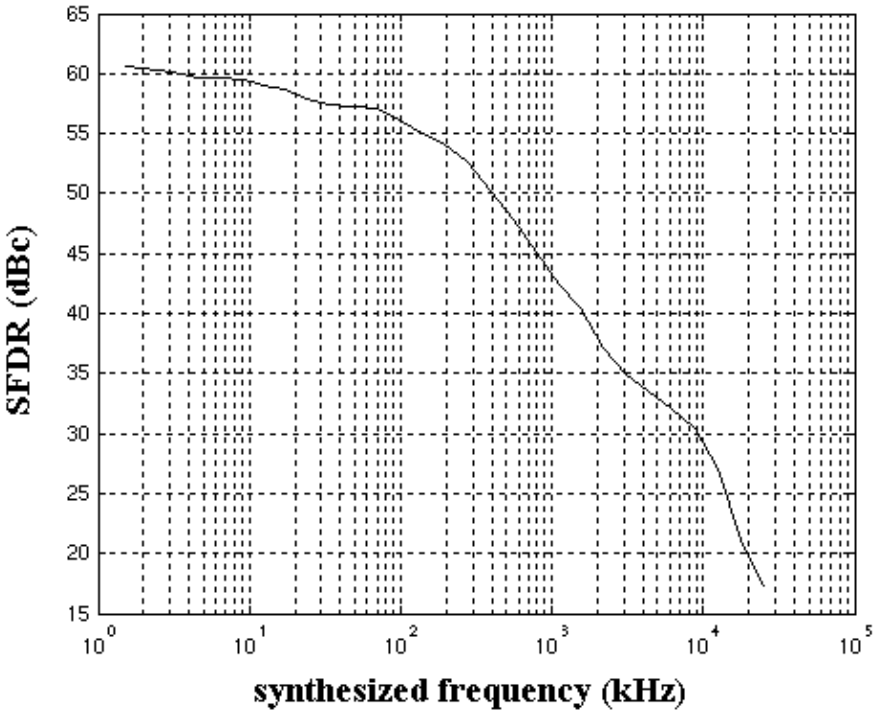


Fig. 4.23 SFDR versus synthesized frequency at f_{clk}=100MHz

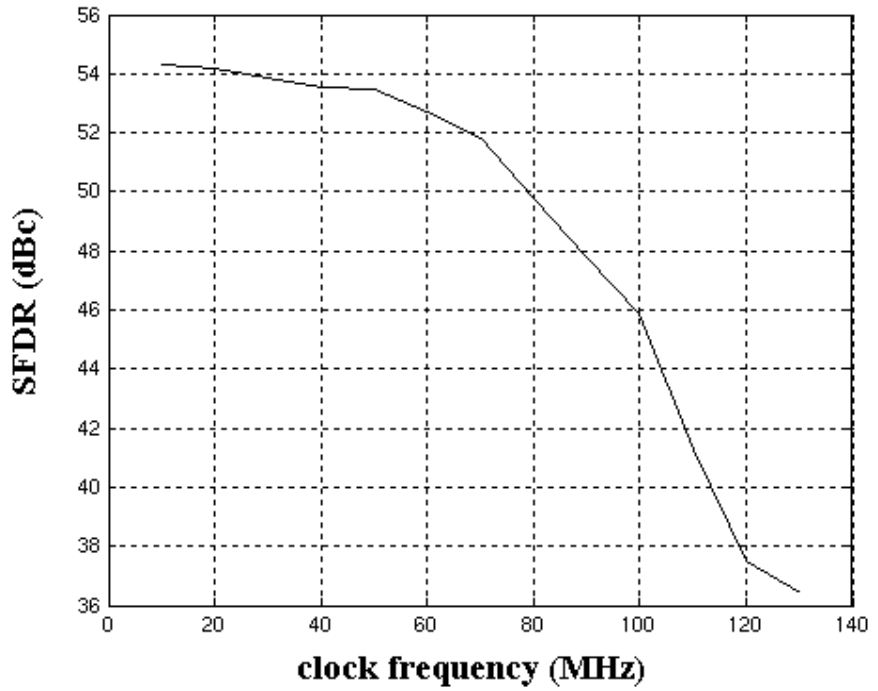


Fig. 4.24 SFDR versus clock frequency at f_{out}= f_{clk}/128

The proposed DDFS is compared with the recently reported synthesizers [30-32] as listed in Table 4.2. Note that the DDFS presented in [30] does not have an On-chip DAC and the DDFS presented in [32] does not have quadrature outputs. The energy of the proposed DDFS (in mW/MHz) is significantly lower than state-of-the-art implementations due to the removal of the ROM and the small DAC size. Therefore, the proposed design is more suitable for low power portable applications. A better SFDR can be achieved, using the proposed architecture, by doubling the number of segments (a = 4). This can increase the SFDR by 12dB at the expense of increasing the power consumption and active area by roughly 50% and 40%, respectively.

Table 4.2 Comparison with recently published work

	[30]	**[31]**	**[32]**	**This Work**
Clock frequency (MHz)/ Operating frequency (kHz)	30 1560	230 200	300 4000	100 200
Phase resolution (bits)	12	10	12	10
Technology (μm CMOS)	0.8	0.5	0.25	0.5
Quadrature outputs	Yes	Yes	No	Yes
On-chip DAC	No	Yes	Yes	Yes
Power dissipation (mW)	9.5	92.5	240	8
Power supply (V)	3.3	3.3	2.5	2.7
SFDR (dBc)	60	55	62	59
Active area (mm^2)	0.9	1.6	1.4	1.4
Energy (mW/MHz)	0.3	0.4	0.8	0.08

4.6. Particular Case

The conventional direct digital synthesizer, i.e., ROM based, can been used to provide sawtooth and ramp waveforms as presented in [34]. The same applies for the proposed DDFS. This is the case because the proposed DDFS share the same front end as the conventional DDFS. The front end is composed of a digital part (the phase accumulator and the complementor) followed by a D/A converter, as shown in Fig. 4.25. Please note that the frequency of the saw-tooth and triangular waveforms, shown in Fig. 4.25, is $4f_{out}$ and $2f_{out}$, respectively, where f_{out} is defined in equation (4.3). The peak of the triangular wave after the linear DAC is V_{max}, while the slope is 1.

One of the possible applications of using the ramp function is in built-in self testing (BIST) [35]. The basic requirement in those kinds of applications is the accuracy of the ramp waveform. This translates directly to the accuracy of the DAC in Fig. 4.25 since the digital part does not play any role in the distortion of the waveform. Many DAC architectures have been reported in the literature [36]. For the resistor string approach, discussed in section 4.3, the accuracy depends on the matching precision of the resistors. Using polysilicon resistors can result in up to 0.1%, i.e., 10 bits of accuracy. This might not be enough for some applications. Dynamic techniques with current switching can be used to realize D/A converters with higher accuracy (up to 16 bits) [36].

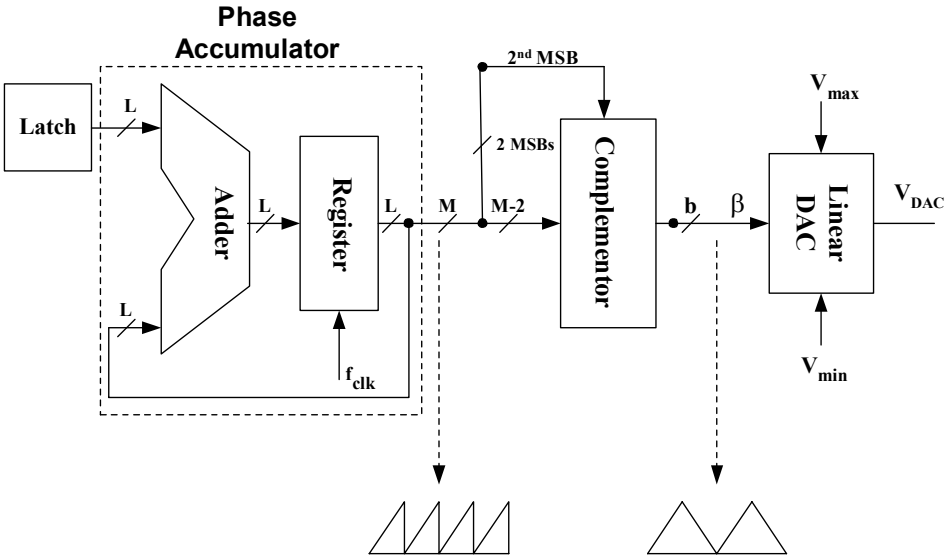


Fig. 4.25 Front end of DDFS

4.7. Conclusions

A low-power ROM-less quadrature DDFS architecture has been presented. It uses a piecewise linear approximation of the sine function. The proposed DDFS has been implemented in 0.5μm CMOS technology and occupies an area of 1.4mm^2. A 16-bit frequency control word results in a tuning resolution of 1.5kHz at a 100MHz clock frequency.

The proposed design operates from a single 2.7V supply while consuming 8mW. The design features an SFDR that is better than 50dBc for synthesized frequencies up to $f_{clk}/256$. The proposed architecture also incorporates different modulation capabilities. The modulation formats include frequency modulation and amplitude modulation. Since the proposed design consumes significantly less power than recently reported designs, it is a good candidate for wireless portable communication applications that use frequency modulation such as Bluetooth and GSM.

There are two frequency limitations in this design. First, higher clock frequencies are limited by speed of the phase accumulator. Digital techniques can be used to solve this problem [31]. Using better process, i.e. smaller feature size, improves the speed of digital blocks such as the accumulator. Second, higher synthesized frequencies are limited by the settling time of the DAC due to large output steps. The values of the resistors in the DAC can be reduced to improve the settling time at the expense of power consumption.

For applications that require higher SFDR, the number of linear segments can be increased at the expense of area and power consumption. For example, doubling the number of segments increases the SFDR by 12dB at the expense of increasing the power consumption and active area by roughly 50% and 40%, respectively. The proposed architecture can be also used to generate triangular and/or saw-tooth waveforms, as shown in Fig. 4.7, using an extra DAC and a smoothing low pass filter.

CHAPTER V

CHAMELEON: A MULTI-STANDARD RECEIVER DESIGN

The differences between Bluetooth and Wi-Fi standard features (data rate, security, communication protocol, etc..) lead to a natural partitioning of applications. However, many electronics devices nowadays, such as laptops, pocket PCs, cell phones, digital cameras, and PDAs, support both Bluetooth and Wi-Fi standards to cover wide range of applications. The cost of supporting both standards is a major concern. Therefore the need for a dual-mode transceiver is evident. To keep the size, and therefore the cost, of such system at a minimum, as many building blocks as possible must be shared in both operating modes. Since Bluetooth and 802.11b share the same RF frequency band, it may be possible to come up with a receiver architecture that can be configured to adopt both standards. RF front end can be shared between both standards. However, since the two standards use different modulation format and have different channel bandwidths, sharing of the baseband blocks is not that easy.

5.1. Possible Receiver Architectures

In section 3.3, we found that low-IF receiver architecture is most suitable for the Bluetooth standard. On the other hand, due to the large channel bandwidth in Wi-Fi, a direct-conversion receiver (DCR) is the best approach. Therefore, building the receiver that supports both standards is not as straightforward. Fundamentally, there are three possible combinations of architecture of both standards when implementing the multi-standard Bluetooth/Wi-Fi receiver:

(1) DCR for Bluetooth and Wi-Fi

(2) Low-IF for Bluetooth and DCR for Wi-Fi

(3) Low-IF for Bluetooth and Wi-Fi

The first combination is the best in terms of area and power consumption. The area is minimal because both standards use the same architecture, which means maximum level of building blocks sharing. However, problems associated with using the DCR architecture for Bluetooth (as discussed in section 3.1) must be solved. In the second combination, each standard uses its best architecture which means easier implementation for each standard at the expense of increased cost because of the reduced sharing of building blocks between both standards. This approach has been used in [37, 38]. The third combination uses the low-IF architecture for both standards. Due to the large Wi-Fi channel bandwidth, IF frequency has to be large enough resulting in high power consumption, especially in Wi-Fi mode [39].

For combination (2), a couple of these solutions are found in literature [37, 38]. The solution [37] uses DCR architecture for Wi-Fi and low-IF architecture for Bluetooth as shown in Fig. 5.1. The RF part (LNA, mixer, and LO) is shared between the two standards while separate baseband parts (Filter, amplifier, and demodulator) are used. An analog multiplexer (SEL) is needed to switch the mixer output either to the Bluetooth or Wi-Fi baseband circuitry. As mentioned before, this approach is not optimum in terms of silicon area. Separate baseband circuitry is used to claim simultaneous operation. However, simultaneous operation is practically not possible using the same RF front-end. This is simply because there might be a situation when there is a strong signal in one standard, requiring low gain in the front-end, and weak signal in the other, requiring high

front-end gain. Furthermore, using the same LO is not acceptable because it is not guaranteed that Bluetooth and Wi-Fi will have the same RF frequency.

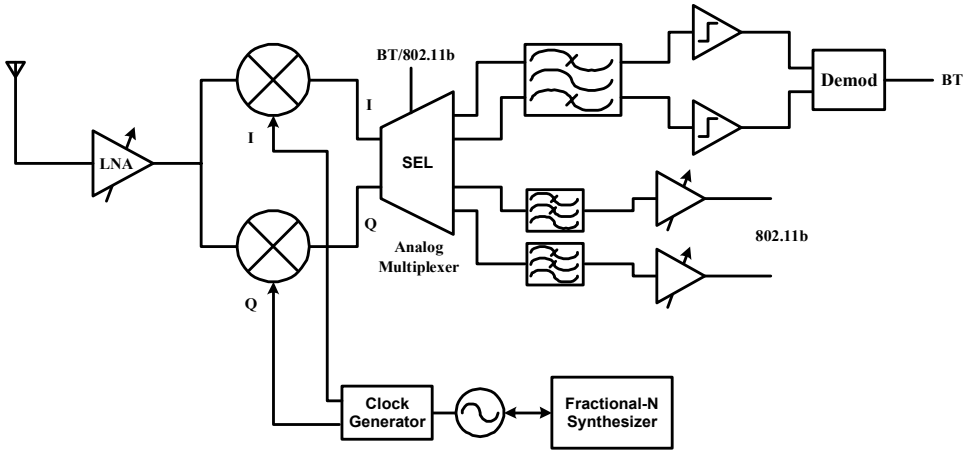


Fig. 5.1 Dual mode receiver architecture used in [37]

The second solution [38] uses double down-conversion architecture as shown in Fig. 5.2. This architecture is used to avoid the LO self-mixing dc offset problem in the receiver and the injection-locking problem between the PA and VCO in the transmitter. The signal at the LNA output is mixed down to 800MHz by the first LO 1.6GHz and then mixed down again to baseband with 800MHz quadrature LOs. In Bluetooth mode, the signal is down-converted to a 2MHz low-IF for channel-select band-pass filtering and limiting. In Wi-Fi mode, the signal is down-converted to dc for channel-select low-pass filtering and signal amplitude control with VGA's. A dual-mode channel-select filter is used to save silicon area. A 5^{th} order Butterworth complex filter with 1MHz bandwidth is used in Bluetooth mode. In Wi-Fi mode, a real filter reconfigured as a fifth order Butterworth LPF with 7.5MHz bandwidth by frequency scaling the filter poles and eliminating the cross-coupling between I and Q branches (see section 3.4.1). Furthermore, the baseband amplifier stage is configured as a four-stage limiter with ac

coupling in Bluetooth mode. In Wi-Fi mode the amplifier stage is configured as a four-stage VGA. Theoretically, due to baseband circuit reuse in both modes, the receiver in [38] should occupy less area than the receiver in [37]. However, since the two modes have different receiver architectures, there is still some area overhead that is needed to reconfigure the baseband blocks for both standards.

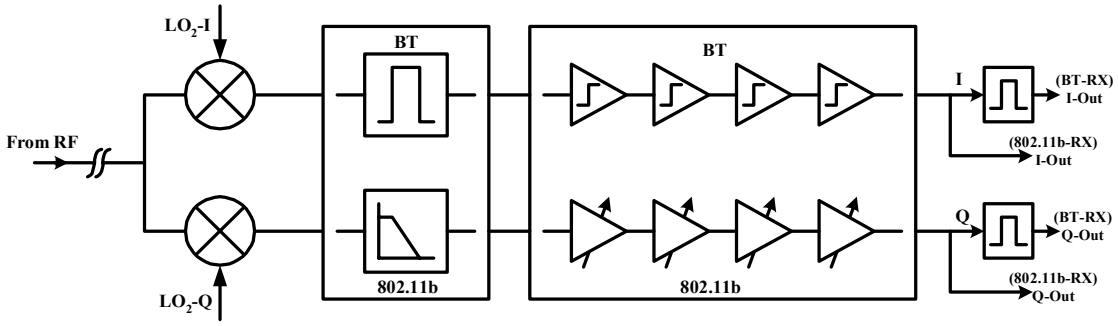


Fig. 5.2 Dual mode receiver architecture used in [38]

5.2. Proposed Direct-Conversion Receiver Architecture

As discussed in the previous section, using the same receiver architecture for both standards allows maximum level of block reuse to save silicon area, which is the main objective of having a dual-mode receiver. The proposed system architecture is shown in Fig. 5.3. The RF Bluetooth/Wi-Fi signal received by the antenna is first amplified by the LNA, and then mixed down with the quadrature output of the local oscillator (LO) to baseband. The quadrature output of the mixer is then passed through a baseband channel select LPF which has tunable bandwidth to suit both standards. The output of the LPF is amplified by the variable gain amplifier (VGA). Since the maximum and minimum signal levels are -4 and -80dBm (in Wi-Fi mode), RF front-end cannot handle the entire signal range with only one LNA gain setting. Therefore, in addition to the VGA gain control,

the LNA has two gain settings, high gain and low gain modes. The LNA and VGA are placed in an automatic gain control (AGC) loop to set the overall receiver gain depending on the signal level at the receiver antenna. The signal level is measured in the digital domain at the ADC output. Therefore, a digitally controlled VGA is desired.

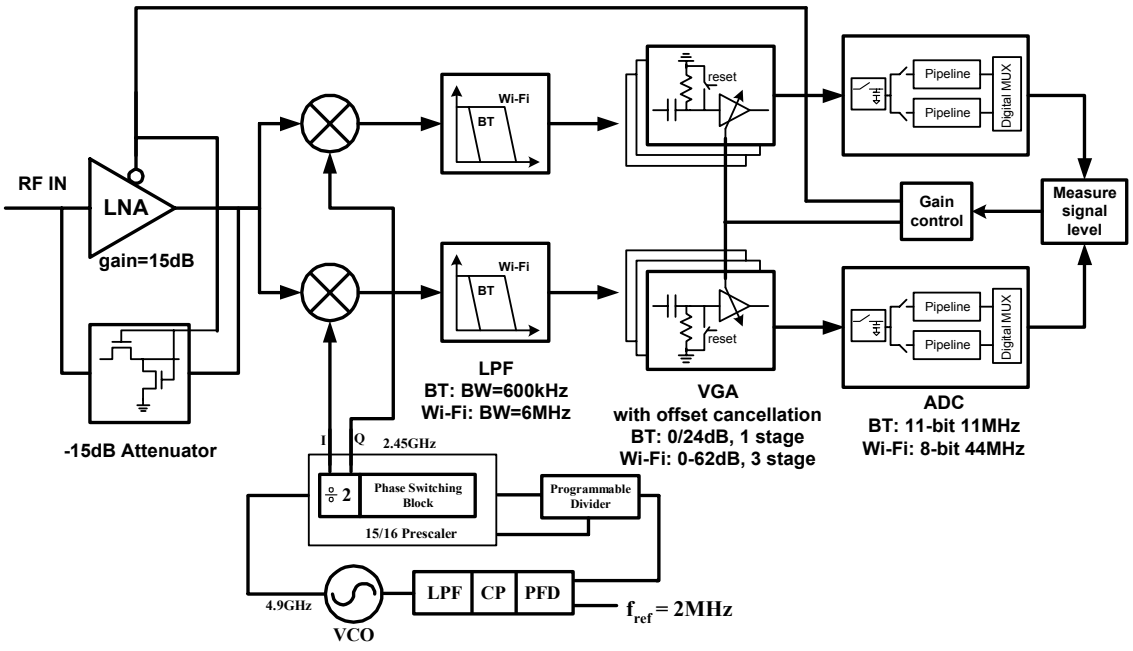


Fig. 5.3 Dual-mode 802.11b/Bluetooth receiver

In Wi-Fi mode, the VGA gain has 2dB steps. Therefore, after AGC settling, the signal level at the ADC input will be adjusted within a couple of decibels, thus relaxing the ADC dynamic range requirement. There are two different preambles and headers defined in Wi-Fi standard [3]: the mandatory supported long preamble and header, which interoperates with the 1Mbit/s and 2Mbit/s DSSS specification (as described in IEEE Std 802.11, 1999 Edition), and an optional short preamble and header. The long and short preamble durations are 144μs and 72μs, respectively. In both short and long preambles, there is enough time to obtain fine gain adjustment in the VGA. On the other hand, in

Bluetooth mode, the preamble time is very short (4μs, which is equivalent to only 4 bits long). Thus, settling time of the AGC loop has to be extremely small, not exceeding 4μs. To alleviate this problem, only two VGA gain settings are used in Bluetooth mode. At start up, the LNA and the VGA are set to the high and low gain modes, respectively. According to the Bluetooth signal level, either the LNA is switched to the low gain mode, the VGA is switched to the high gain mode, or the LNA and VGA gains are kept as they are. Therefore, a maximum of only one gain adjustment step is made in Bluetooth mode. As a result, the signal level at the ADC input can vary by about 30dB. This increases the required ADC resolution by about 5 bits.

5.3. System Design Issues

In the following few subsections, system design issues in DCR architectures are discussed [12]. The goal is to determine the optimal specifications for the system and for each building block System and blocks specifications are derived in section 5.5. The discussions are supported with simulation results that are performed using System View and MATLAB. System View has a library for 802.11b standard that we used mainly to examine the effect of the receiver non-idealities on the bit error rate (BER) performance of the demodulator. Fig. 5.4 shows the basic simulation setup for the Wi-Fi simulations using *System View*. On the other hand, MATLAB is used to assess the Bluetooth demodulator performance in the presence of these non-idealities. A MATLAB code (appendix B) was developed to simulate the effects of the receiver non-idealities on the BER performance of the Bluetooth GFSK optimal non-coherent detector. Block diagram of the Bluetooth GFSK demodulator is depicted in Fig. 5.5. The impulse response of the

Gaussian filter used in Bluetooth extends from $-T_b$ to T_b (T_b is the bit duration) as shown in Fig. 2.3. This means the intersymbol interference (ISI) occurs with the previous and the following bits. To make the demodulator immune against the effect of ISI, eight matched filters are used for all the combinations of the three bits (the current, previous, and the following bits).

5.3.1. BER versus SNR

In order to evaluate the effect of each of the receiver non-idealities, we need to know the minimum SNR to meet the required BER in the ideal case for each standard. In this simulation, the signal is assumed to be perfectly synchronized with the demodulator correlator. Since a digital demodulator is used for each mode, the correlator performance depends on the sampling rate. Higher sampling rate means better demodulator performance, but the simulation time is longer. To have a reasonable accuracy of the simulation results compared to the ideal case, the sampling rates used are 20MSample/s and 88MSample/s for Bluetooth and Wi-Fi modes, respectively. Figs. 5.6 and 5.7 show the ideal case BER-SNR curves for Bluetooth and Wi-Fi modes where it is seen that the required SNR_{min} is 12.25 and 11.4dB, respectively.

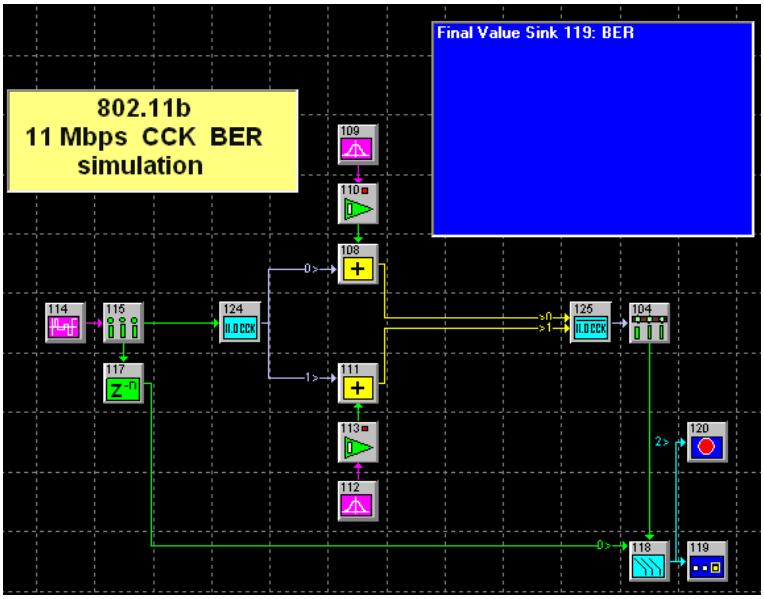


Fig. 5.4 Basic Wi-Fi simulation setup in *System View*

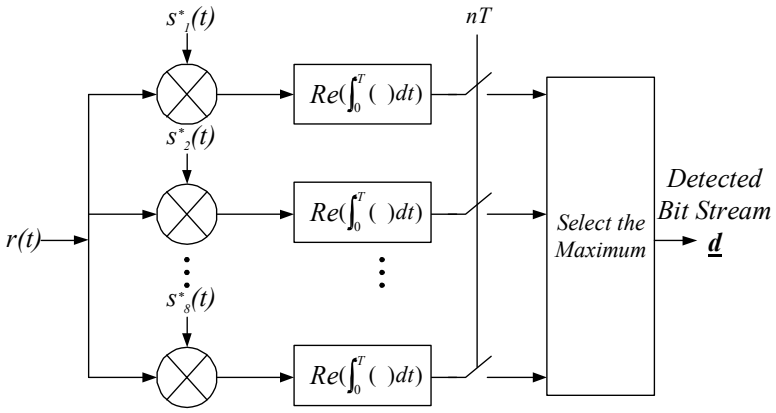


Fig. 5.5 Bluetooth GFSK non-coherent demodulator used in MATLAB

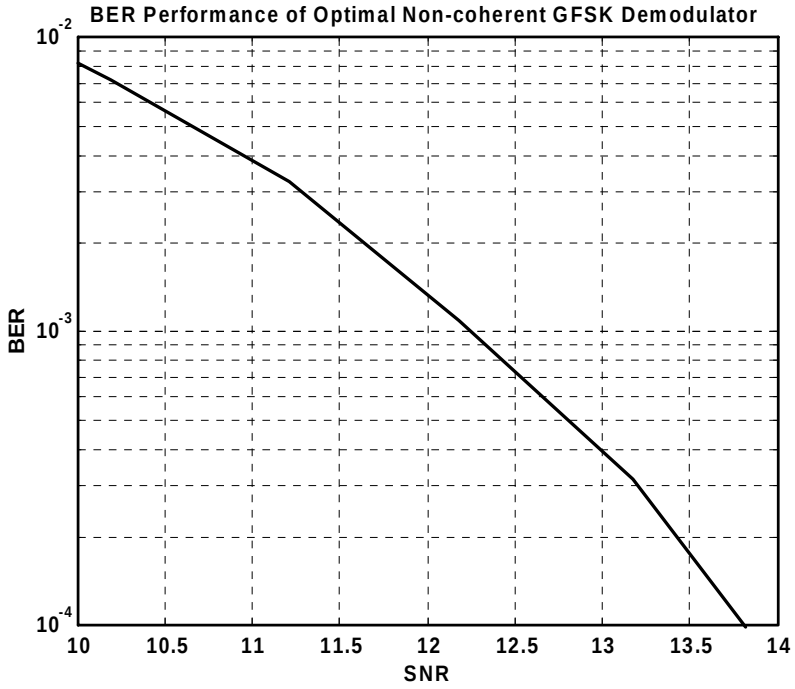


Fig. 5.6 BER versus SNR in Bluetooth mode in the ideal case
(Sampling rate = 20MSample/s)

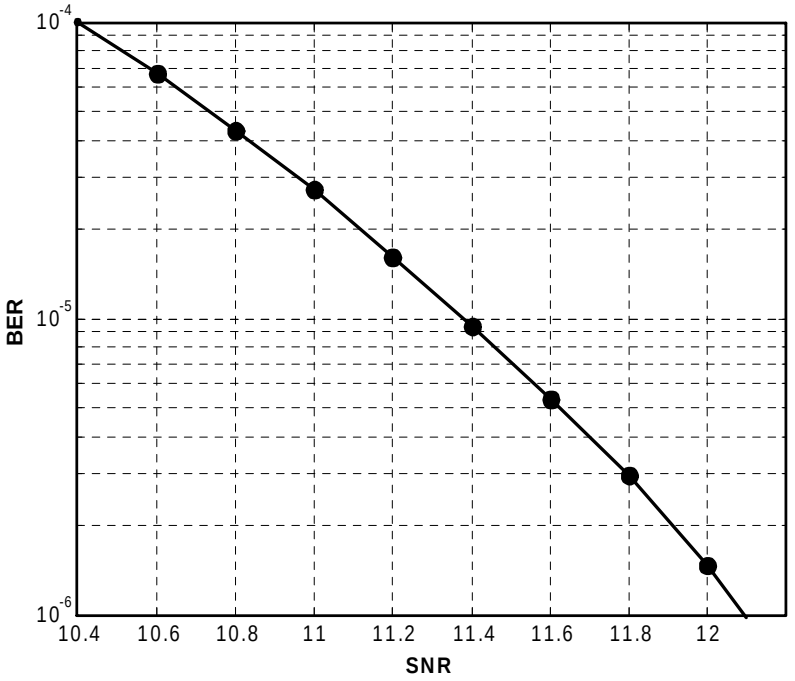
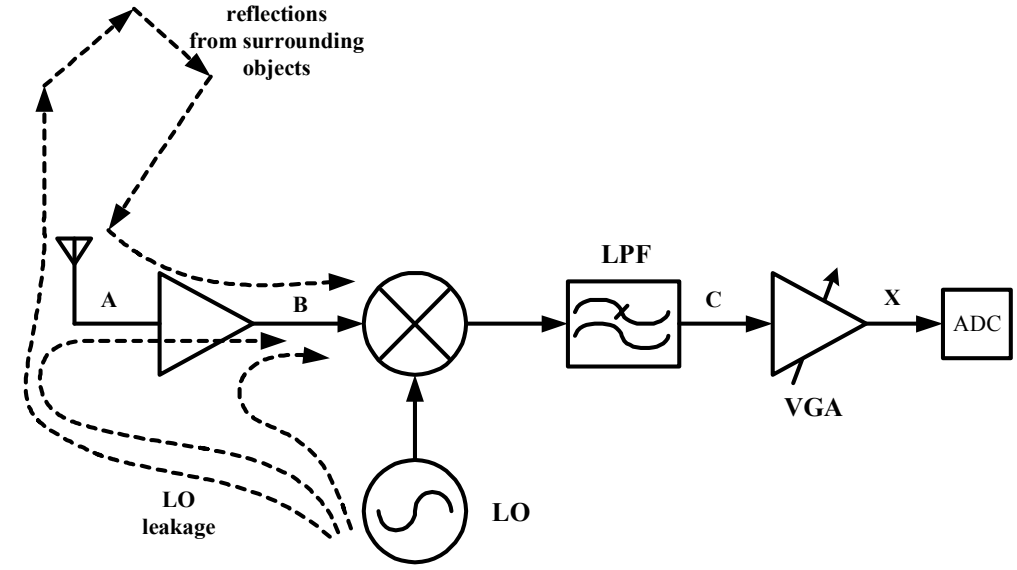


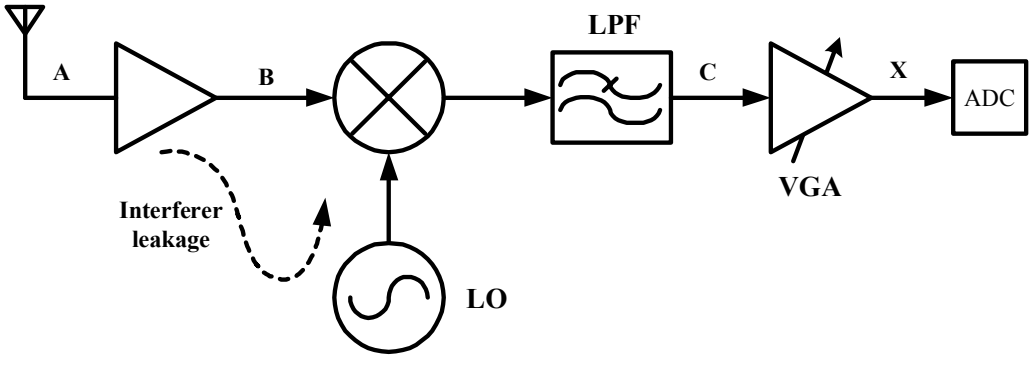
Fig. 5.7 BER versus SNR performance of Wi-Fi 11Mbit/s CCK demodulator
(Sampling rate = 88 MSample/s)

5.3.2. DC Offset Problem

Since in DCR architecture the downconverted band extends to zero frequency, extraneous offset voltages can corrupt the signal, and more importantly, saturate the following stages. To understand the origin and impact of offsets, consider the receiver shown in Fig. 5.8, where the LPF is followed by an amplifier and an analog-to-digital converter (ADC). Let us make two observations.



(a)



(b)

Fig. 5.8 Self-mixing of (a) LO (b) Interferers

First, the isolation between the LO port and the inputs of the mixer and the LNA is not perfect. In other words, finite amount of feedthrough exists from the LO port to points A and B (Fig. 5.8(a)). This effect is called "LO leakage" and it arises from the capacitive and substrate coupling, and if the LO signal is provided externally, the leakage comes from the bond wire coupling. The leakage signal appearing at the inputs of the LNA and the mixer is now mixed with the LO signal, thus producing dc offset at point C. This phenomenon is called "self mixing". A similar effect occurs if a large interferer leaks from the LNA or mixer input to the LO port and is multiplied by itself (Fig. 5.8(b)).

Second, the total gain from the antenna to point X is typically around 80-100dB so as to amplify the microvolt input signal to a level that can be digitized by a low cost, low power ADC. Of this total gain, typically 25-35dB is contributed by the LNA/mixer combination. The offset at the mixer input due to LO coupling to the LNA input can be expressed as:

$$\text{Output offset} = LO\ power + (LO\ to\ LNA\ coupling) + G_{LNA} + G_{Mixer} \quad (5.1)$$

With the above observations, we can obtain a rough estimate of the offset resulting from self-mixing to appreciate the problem. Suppose in Fig. 5.8(a) the LO has a peak-to-peak swing of 0.63V (≈ 0dBm for R = 50Ω)[3] and experiences an attenuation of 60dB as it couples to point A. If the gain of the LNA/ mixer is 30dB, then the offset produced at the output of the mixer is -30dBm, which is equivalent to 10mV. We also note that the desired signal level at this point can be as low as $30\mu V_{rms}$. Thus, if directly amplified by VGA gain (could be up to 50-60dB), the offset voltage saturates the VGA stages.

[3] To convert from volts to dBm, we use the formula: $P(in\,dBm) = P(in\,dBW) + 30$, where $P(in\,dBW) = 10\log(\frac{V_{P-P}^{2}}{8R})$

The offset problem is exacerbated if self-mixing varies with time. This occurs when the LO leaks to the antenna and is radiated and subsequently reflected from moving objects (e.g. a car moving at high speed) back to the receiver.

DC offset can also be generated due to mismatches between circuit components. Typical CMOS transistor V_T mismatch can be in the order of few millivolts. Therefore, Offsets generated by the mixer and filter can also saturate the VGA stages.

We infer from the above discussion that DCR architecture necessarily requires some means of DC offset removal. Depending on the DC offset removal approach, residual dc offset might appear at the ADC input. If not removed in the digital domain, this DC offset may cause degradation in BER performance of the Bluetooth and Wi-Fi demodulator. Figs. 5.9 and 5.10 show the demodulator degradation due to DC offsets in Bluetooth and Wi-Fi modes, respectively. In order to have less than 0.3dB degradation in the required SNR, the residual offset at the demodulator input should be less than 10% (relative to the peak signal value) in Bluetooth and 5% in Wi-Fi.

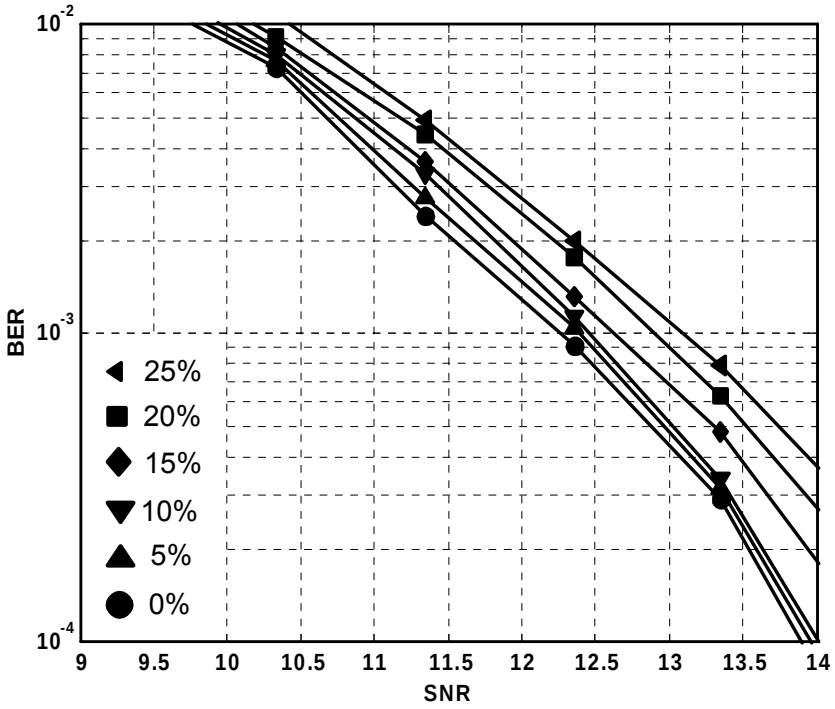


Fig. 5.9 BER performance of Bluetooth GFSK demodulator for different DC offsets (percentage relative to the peak signal value)

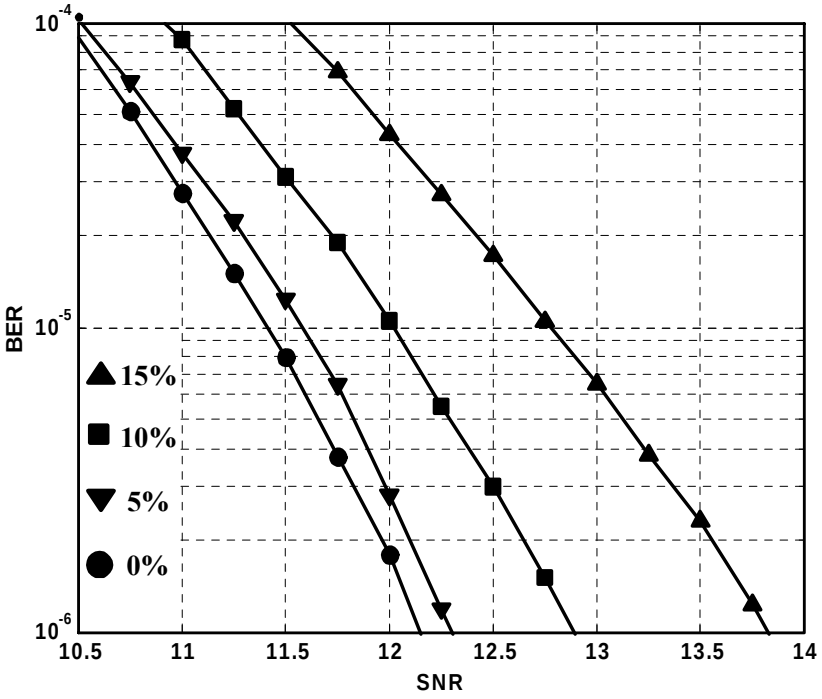


Fig. 5.10 BER performance of Wi-Fi 11Mbit/s CCK demodulator for different DC offsets (percentage relative to the peak signal value)

It should be mentioned that no offset compensation scheme is implemented in the demodulator for the simulations results shown in Figs. 5.9 and 5.10. Degradation due to dc offset can be reduced by digitally compensating the residual offset. Two possible approaches to eliminate the dc offset are discussed in the next two subsections.

5.3.2.1. AC coupling

A possible approach to remove the dc offset is to employ, i.e., passive RC high-pass filtering, in the downconverted signal path [40]. However, the spectra of the Bluetooth and Wi-Fi baseband signal exhibit a peak at dc as shown in Figs. 5.11 and 5.12. Such signals are corrupted if filtered with a high corner frequency. Simulations using MATLAB and System View were conducted to measure the BER performance degradation due to AC coupling. As will be described in section 6.4, four first-order HPFs are used in the VGA, one in front of each of the three VGA stages and one at the output of the last stage. In Bluetooth mode, since only one VGA stage is used, the signal is passed through the last two HPFs. On the other hand, since there are three VGA stages used in Wi-Fi mode, the signal is passed through the four HPFs. The effects of these HPFs were included in the simulations. Fig. 5.13 shows the Bluetooth GFSK demodulator BER simulation using MATLAB with two cascaded HPF with the same cutoff frequency. The degradation is less than 0.1dB if the cutoff frequency is less than 1kHz. In Wi-Fi CCK demodulator simulation using System View with four cascaded HPFs. The cutoff frequency of the last two HPFs is set to 1kHz as required in the Bluetooth mode. The cutoff frequency of the first two HPFs is swept in the simulation. Fig. 5.14 shows that the degradation in CCK demodulator performance is less than 0.2dB if the cutoff frequency of the first two HPFs is less than 5kHz.

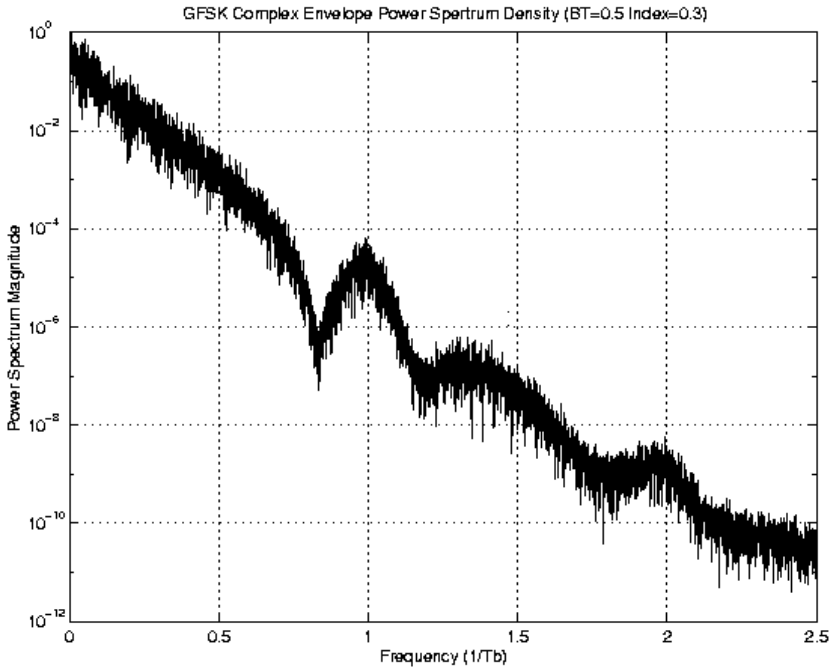


Fig. 5.11 Power spectral density of Bluetooth GFSK signal

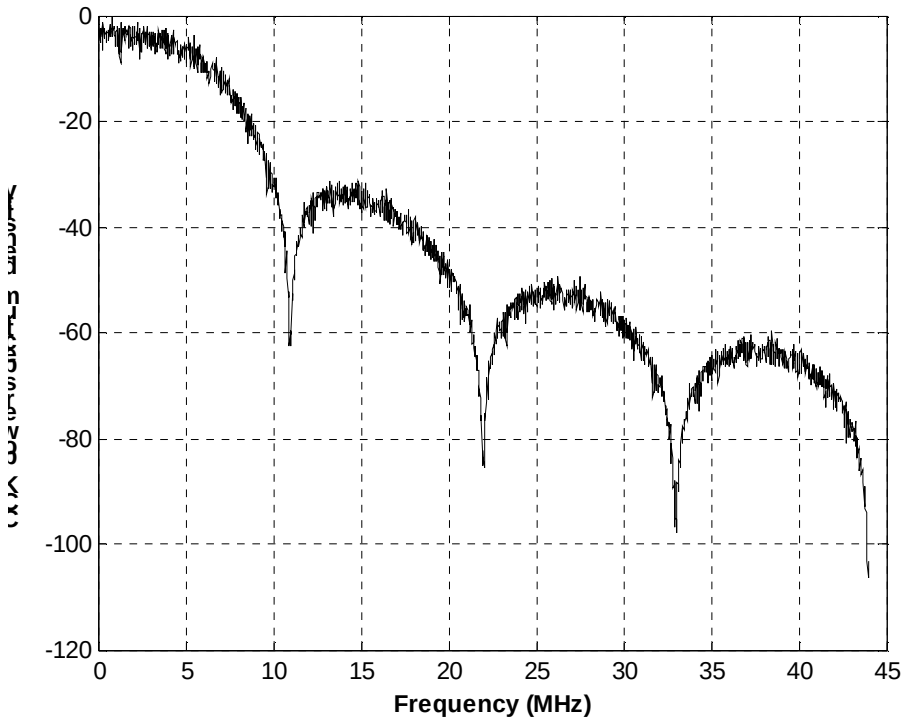


Fig. 5.12 Power spectral density of Wi-Fi CCK signal

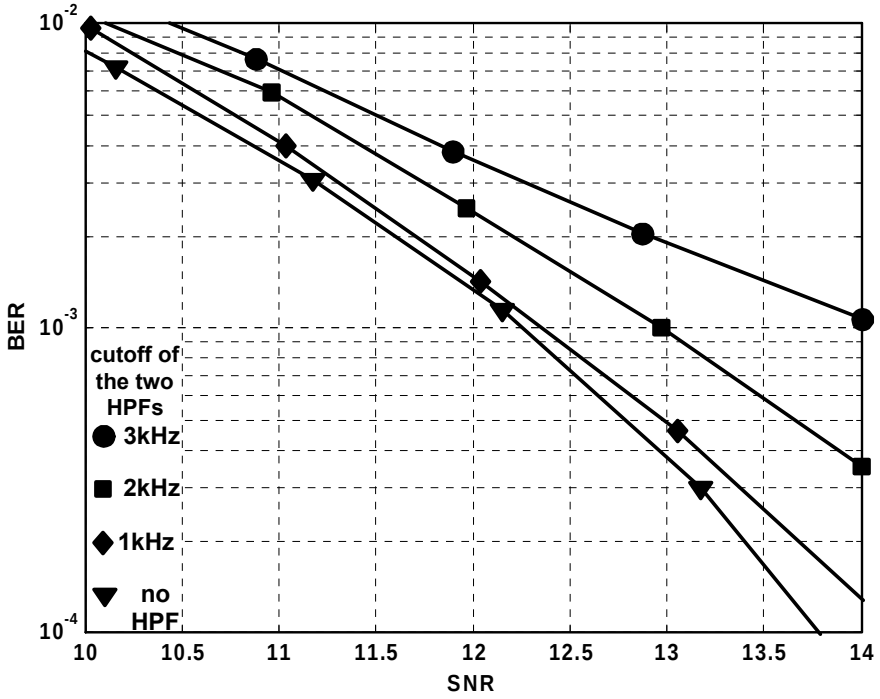


Fig. 5.13 Bluetooth GFSK demodulator performance with two HPFs

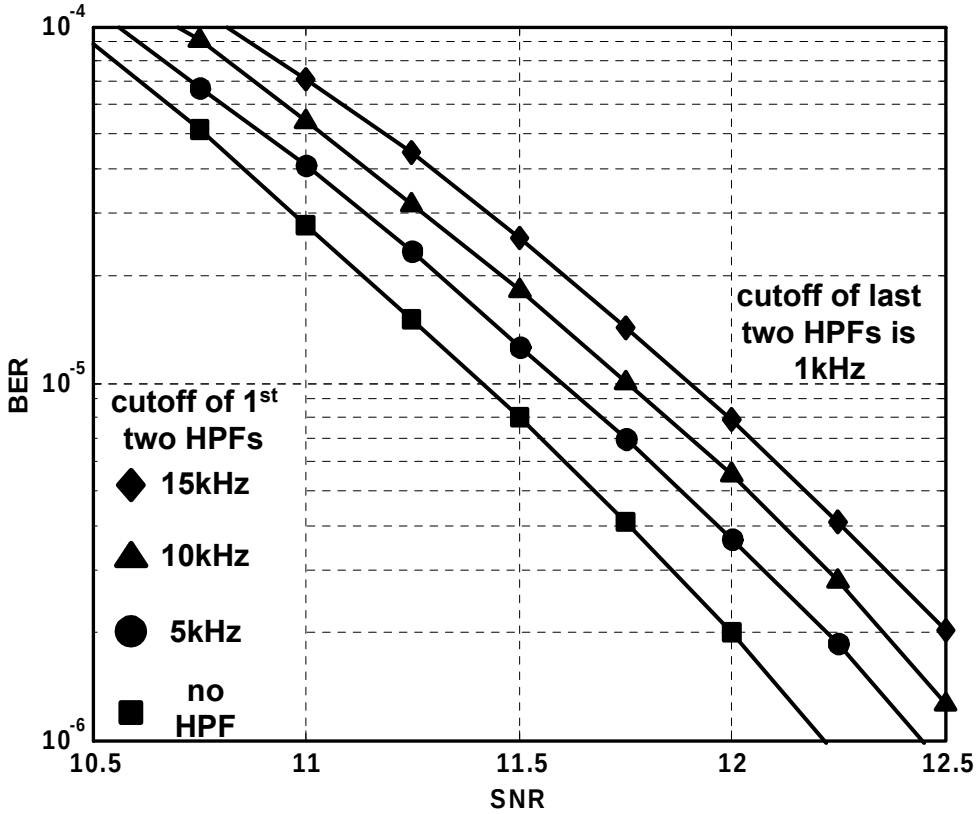


Fig. 5.14 Wi-Fi 11Mb/s CCK demodulator performance with four HPFs

Another issue concerning using passive RC HPF to remove the offset is the thermal noise of the resistor R. The integrated thermal noise at the output of the HPF is 2kT/C (the factor 2 is due to differential operation), where k is Boltzmann constant and T is the absolute temperature. If the received signal is at -85dBm (12.6μV_{rms}), and assuming the total gain of the LNA, mixer, and LPF is 37dB, the differential signal level at the VGA input will be -85 + 37 = -48dBm (0.89mV_{rms}). If the first AC coupling stage is placed at the VGA input, the two differential capacitors must at least 1pF each so that 2kT/C remains 20dB below the signal level. AC coupling capacitors at later VGA stages can be smaller because the signal level is larger. The low frequency part of the kT/C noise spectrum of the first HPF stage is partially rejected by the later HPF stages.

5.3.2.2. Offset cancellation loop

By sensing the offset at the ADC input, integrating it, and then injecting the correction signal at the input of the VGA (a point where the offset is not too high to affect the circuit operation) as shown in Fig. 5.15 [41, 42]. If the gain of the VGA is K, the transfer function of the closed loop system shown in Fig. 5.15 is expressed as:

$$\text{Closed loop response} = \frac{K}{1 + \frac{\omega_o}{s} K} = K \frac{s}{s + \omega_o K} \tag{5.2}$$

Therefore, the overall response of the closed loop VGA with the offset cancellation feedback loop is a HPF response and, like AC coupling, it still corrupts the signal spectrum at low frequencies. Note that in equation (5.2), the HPF cut-off frequency depends on the integrator time constant and the loop gain (including the VGA gain). Therefore, the cutoff frequency of the HPF response will vary with the VGA gain, unless this gain variation is compensated in the feedback gain such that $\omega_o K$ is kept constant.

Thus, this approach is more complex than AC coupling. Alternatively, as depicted in Fig. 5.16, the DC offset can be sensed in the digital domain at the output of the ADC and, therefore, requiring a digital-to-analog converter (DAC) to inject the correction signal at the VGA input [43]. Unlike pure analog approaches, the injected correction signal is available digitally at the DAC input and it can be stored forever. Therefore, once the proper correction digital word at the DAC input is found, the loop can be opened and the correction digital word is stored and applied to the DAC input. The main advantage of this approach is that the signal is dc coupled through the VGA and the low frequency portion of the signal spectrum is not corrupted. However, this approach is rather more complex than AC coupling or analog offset cancellation loop. It requires high DAC resolution (resolution increase with VGA maximum gain). In addition, the DAC output swing has to be as low as the dc offset range at the injection point which is usually which is usually in the order of tens of millivolts.

Offset cancellation scheme using ac coupling HPFs has been used in the dual mode receiver. The scheme is described in detail in chapter VI.

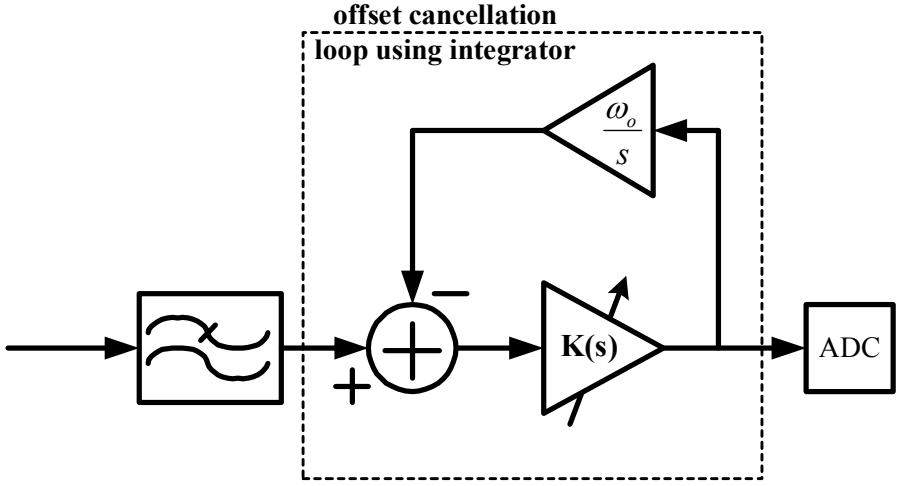


Fig. 5.15 Analog offset cancellation loop using feedback integrator [41, 42]

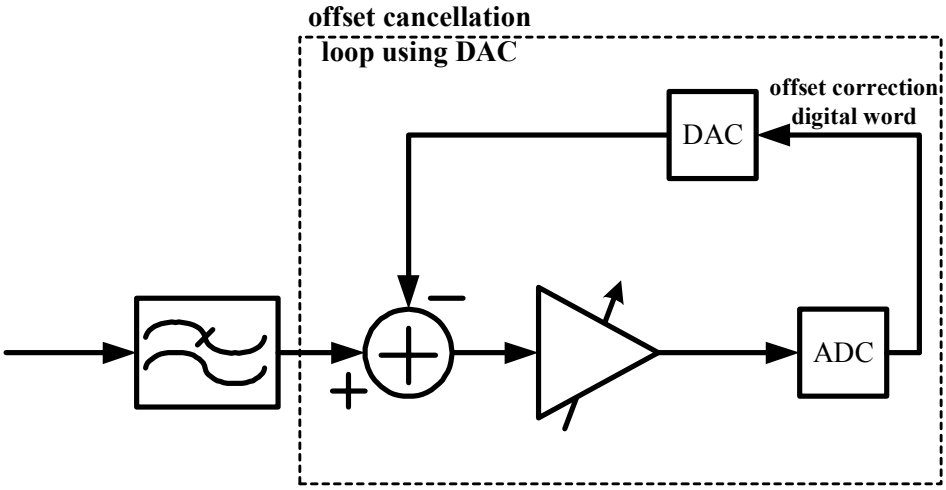


Fig. 5.16 Offset cancellation loop using DAC [43]

5.3.3. Filter Frequency Response

Several simulations have been done to assess the effect of filter approximation and bandwidth on the BER performance demodulator in both modes. Fig. 5.17 shows the Bluetooth demodulator performance using 5^{th} order Butterworth LPF and 4^{th} Chebychev LPF with 0.5dB pass-band ripples. Both filters have 600 kHz bandwidth. There is more than 0.5dB degradation when using Chebychev approximation compared to Butterworth approximation. This can be attributed to the fact that Chebychev filter has more group delay variation in the pass-band than in Butterworth. Therefore, a Butterworth approximation is chosen for this receiver. The next step is to determine the optimum filter bandwidth. The higher the filter bandwidth is, the less attenuation for adjacent channel interferer and the more noise passed to the demodulator. The lower the bandwidth is, the more significant portion of the signal spectrum is rejected. Fig. 5.18 shows the Bluetooth BER performance using Butterworth filter with different bandwidths. The optimum Butterworth filter bandwidth is 600kHz in Bluetooth mode.

In Wi-Fi mode, on the other hand, 5th order Butterworth filter with a bandwidth of 6MHz has superior BER performance over a 4th order Chebychev with the same bandwidth as depicted in Fig. 5.19. Fortunately, 5th order Butterworth is the optimum filter approximation for both standards. Therefore, there is no need to change the relative poles locations when switching between the standards and only frequency tuning is required. The optimum Butterworth filter bandwidth is obtained from the simulation plotted in Fig. 5.20. 6MHz appears to be the bandwidth corresponding to best BER performance. However, simulations showed that the BER performance does not degrade significantly when the cutoff frequency is changed by ±500 kHz; relaxing the filter frequency tuning accuracy.

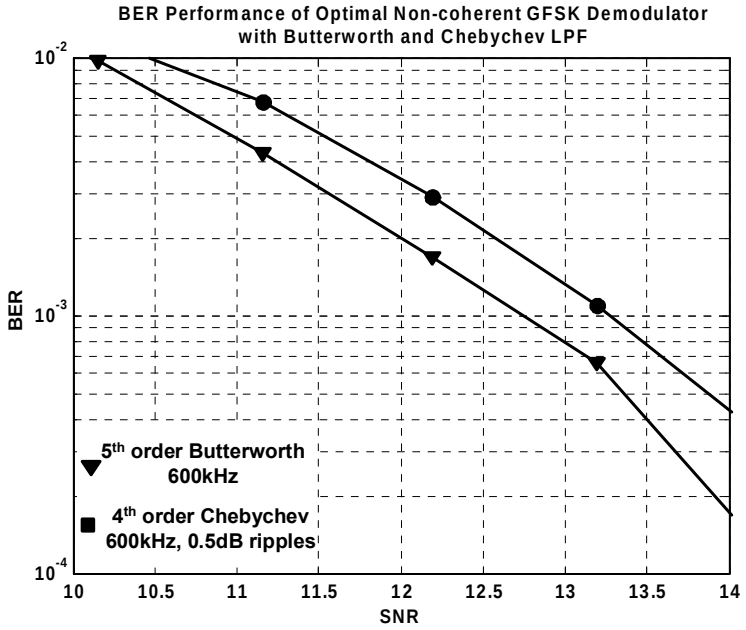


Fig. 5.17 BER performance using different LPF approximations in Bluetooth mode

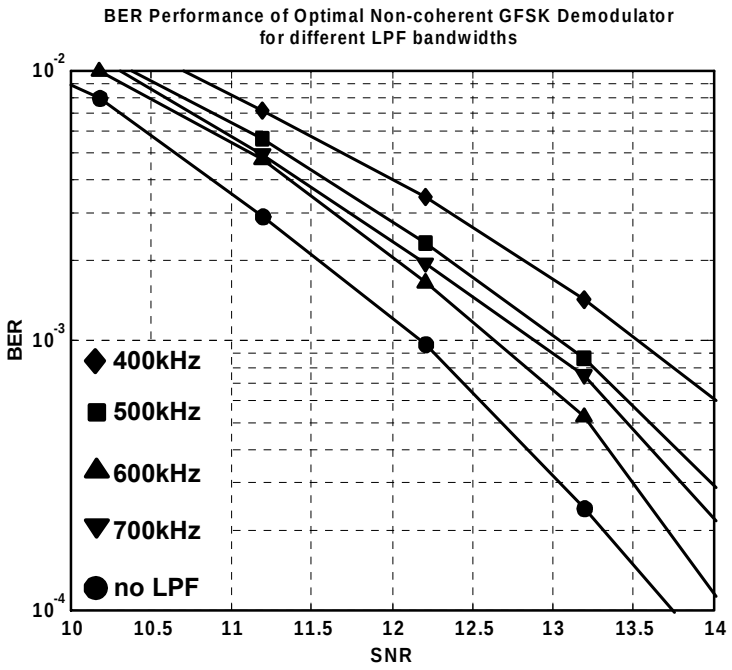


Fig. 5.18 Bluetooth BER performance using Butterworth LPF with different bandwidths

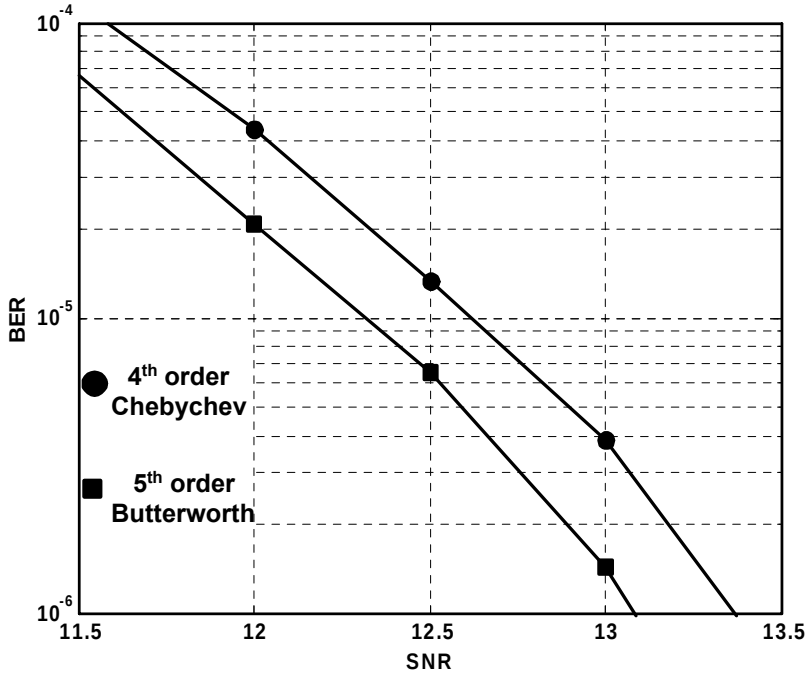


Fig. 5.19 Wi-Fi 11Mbit/s CCK BER using different LPF approximations

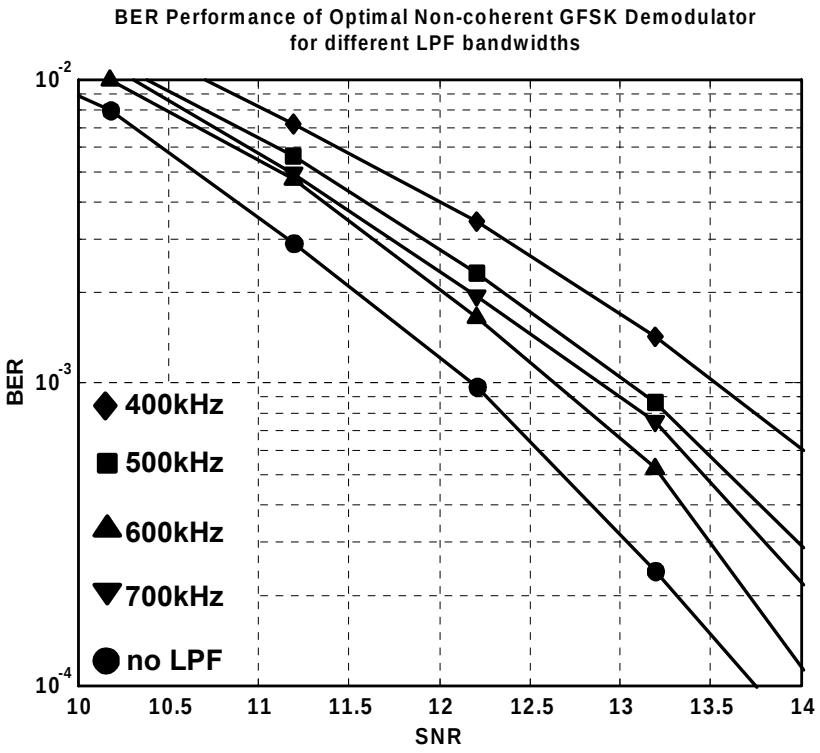


Fig. 5.20 Wi-Fi BER performance using Butterworth LPF with different bandwidths

5.3.4. ADC Resolution and Sampling Rate

To avoid degrading the SNR, the ADC quantization noise should be well below the noise floor at the VGA output. In other words, the ADC dynamic range should be higher than the required SNR_{min} which is about the same in both modes. Fig. 5.21 shows the degradation due to ADC quantization in Bluetooth mode. The degradation is about 1dB for 3bit ADC, which is expected since it has only 18dB of dynamic range, not sufficiently higher than SNR_{min}. With 4bit quantization, degradation is only 0.2dB degradation relative to the ideal case with no quantization. But the signal level at the ADC input can vary by about 26dB. To cover this variation, the ADC resolution should be increased by about 5bits. So the minimum required ADC resolution is 9bits in Bluetooth mode. To allow for some margin for implementation loss and signal dynamic range, 11bits ADC is

designed. Fig. 5.22 shows the effect of the sampling rate on the BER performance of the Bluetooth demodulator. A lower sampling rate leads to higher timing errors between the incoming signal and the demodulator matched filter and, therefore, greater loss in SNR_{min}. Fig. 5.22 shows that the degradation is less than 0.1dB for sampling rates of 8MHz or higher. A sampling rate of 11MHz is used in the Bluetooth mode. This frequency is chosen because it can be derived easily from the 88MHz crystal oscillator.

In Wi-Fi mode, the degradation due to ADC quantization is depicted in Fig. 5.23. The degradation is less than 0.1dB for 5 bits of resolution. However, 8bits are implemented in Wi-Fi mode to account for AGC resolution, possible signal level variations, error in signal level measurement. Fig. 5.24 shows the effect of ADC sampling speed on the Wi-Fi BER performance. 44MHz appears to be a good compromise between ADC power consumption and BER degradation.

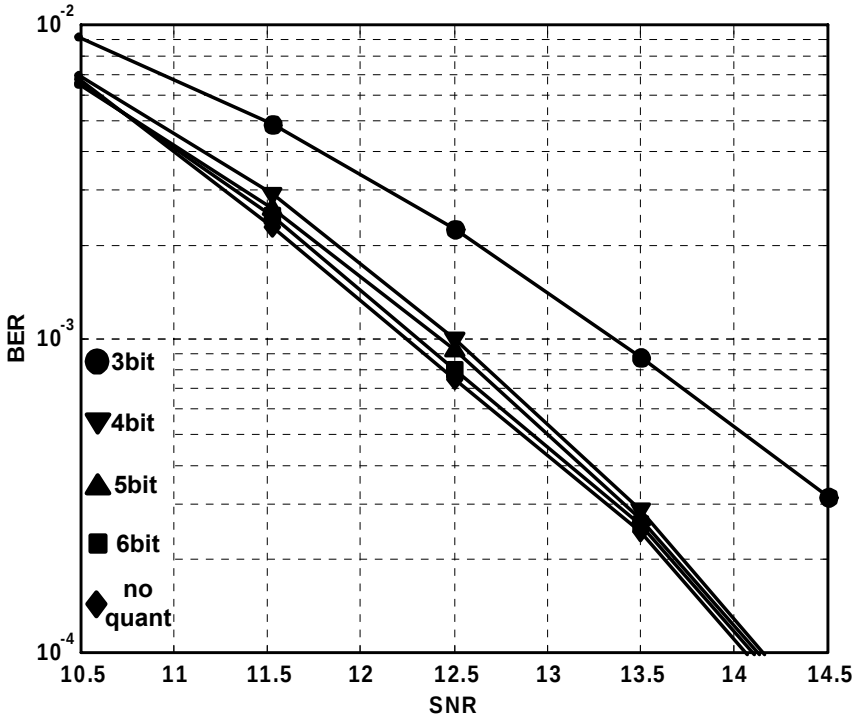


Fig. 5.21 Bluetooth GFSK BER performance for different ADC quantization bits

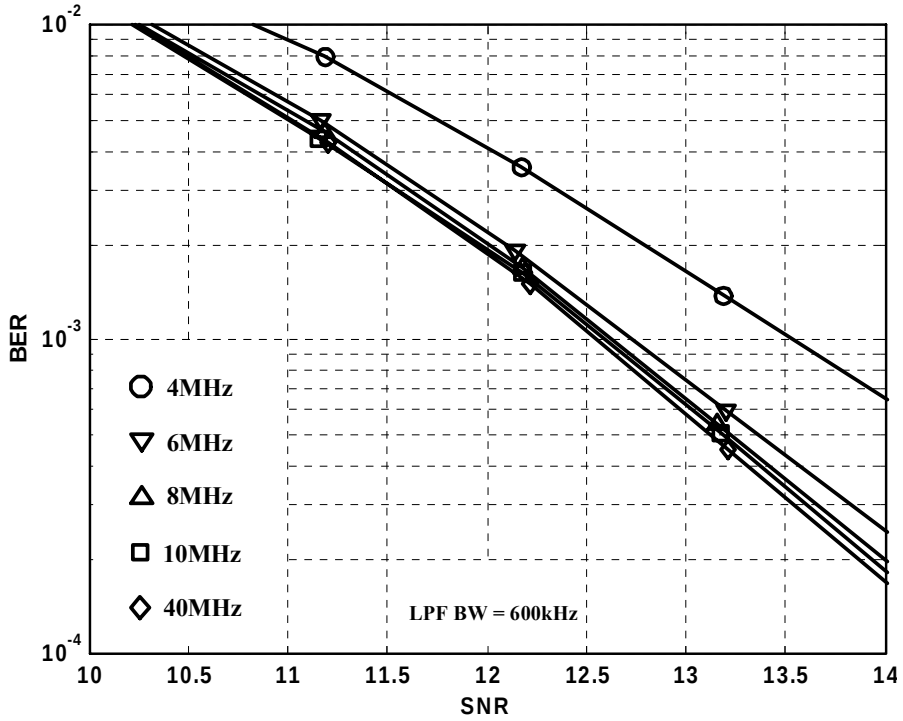


Fig. 5.22 Bluetooth GFSK BER performance for different ADC sampling rates

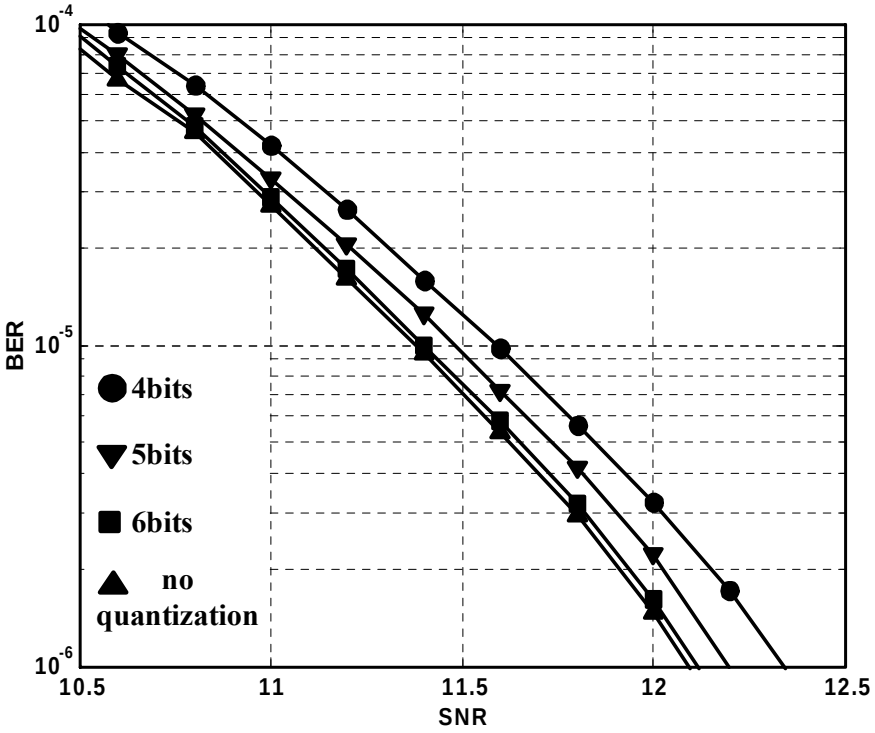


Fig. 5.23 ADC quantization effect on BER performance of Wi-Fi 11Mb/s CCK

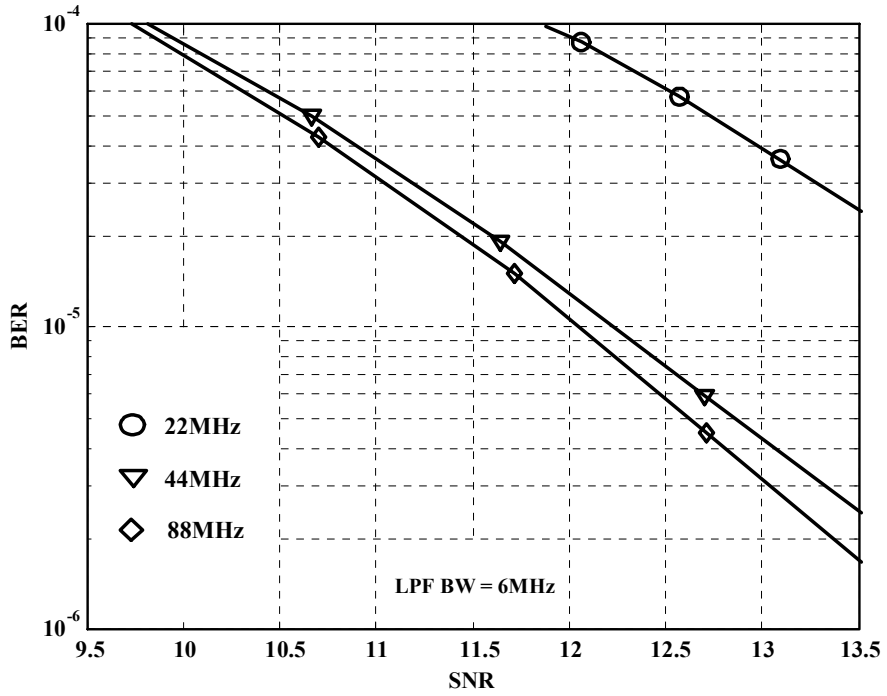


Fig. 5.24 Wi-Fi 11Mb/s CCK BER performance for different ADC sampling rates

5.3.5. VGA Dynamic Range

The dynamic range of the VGA is determined by the signal power level range at the receiver antenna. In Wi-Fi, the minimum (sensitivity) and maximum signal levels at the receiver antenna are specified to be -80dBm to -4dBm for the 2Mbit/s data rate. To cover this 76dB variation in the signal level, the overall gain of the receiver must be adjusted such that, for all signal levels, no block is saturated and the signal level at the ADC input is within the allowable range to demodulate the signal. To avoid saturating the RF front-end at high signal levels, the LNA is bypassed and an attenuator is used. The LNA/attenuator stage provides 30dB gain step. Therefore, reducing the signal level dynamic range to 76-30 = 46dB at the blocks following the LNA. In Wi-Fi, signal level at the input of the ADC has to be well defined to make the most out of the 8-bit ADC resolution. A VGA with 2dB gain step is used to reduce the signal dynamic range at the

ADC input to 2dB. To have some margin in sensitivity and maximum signal levels specifications, a 0-62dB VGA is designed with a total of 16dB margin at the high and low ends. This large margin is also used to compensate for possible gain errors in the receiver blocks due to process variations and non-ideal components, especially in the RF front end, and to achieve better sensitivity and maximum signal levels.

In Bluetooth mode, due to the very short time allowed for the preamble, the settling time of the AGC loop has to be extremely short. Therefore, only two VGA gain settings are used (0/24dB using the 3rd VGA stage only). Upon reception of the Bluetooth preamble, the LNA (VGA) is set to maximum (minimum) gain; then, a maximum of one gain control step is needed to adjust the receiver gain, minimizing the overall receiver settling time. Due to the large gain step employed in Bluetooth mode, the signal level at the output of the VGA may vary by about 30dB, and the ADC must be designed to accommodate this dynamic range for the Bluetooth signal.

5.3.6. I/Q Mismatches

In low-IF receivers [15, 44], gain and phase mismatches between I and Q channels affect the image rejection ratio of the receiver. Although the image problem is eliminated in DCR architecture, the gain and phase mismatches still affect the BER performance of the demodulator. However, matching requirements in DCR is not as tough as in low-IF architecture. Effects of gain and phase mismatches on the Bluetooth BER performance are depicted in Figs. 5.25 and 5.26, respectively. For less than 0.2dB loss in SNR_{min}, gain and phase mismatches should be less than 10% and 5°, respectively. The Wi-Fi BER degradations due to gain and phase mismatches are plotted in Figs. 5.27 and 5.28. SNR_{min} degradation is below 0.1dB for gain and phase mismatches 5% and 5°, respectively.

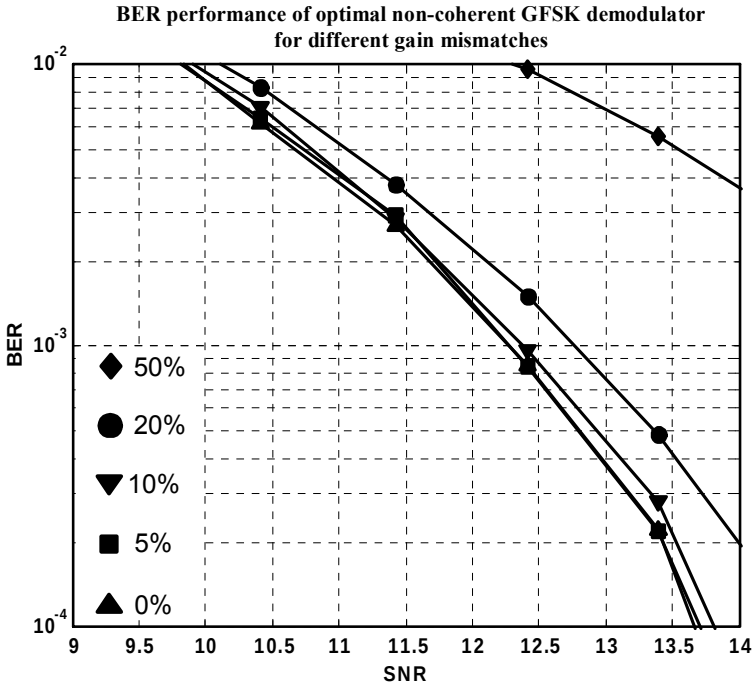


Fig. 5.25 BER performance of Bluetooth demodulator for different gain mismatches

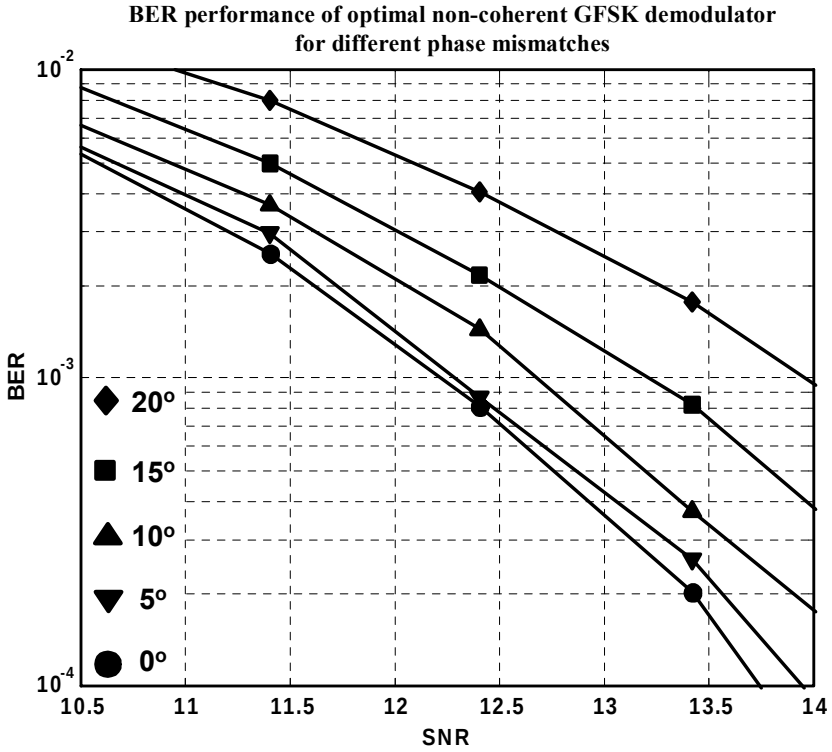


Fig. 5.26 BER performance of Bluetooth demodulator for different phase mismatches

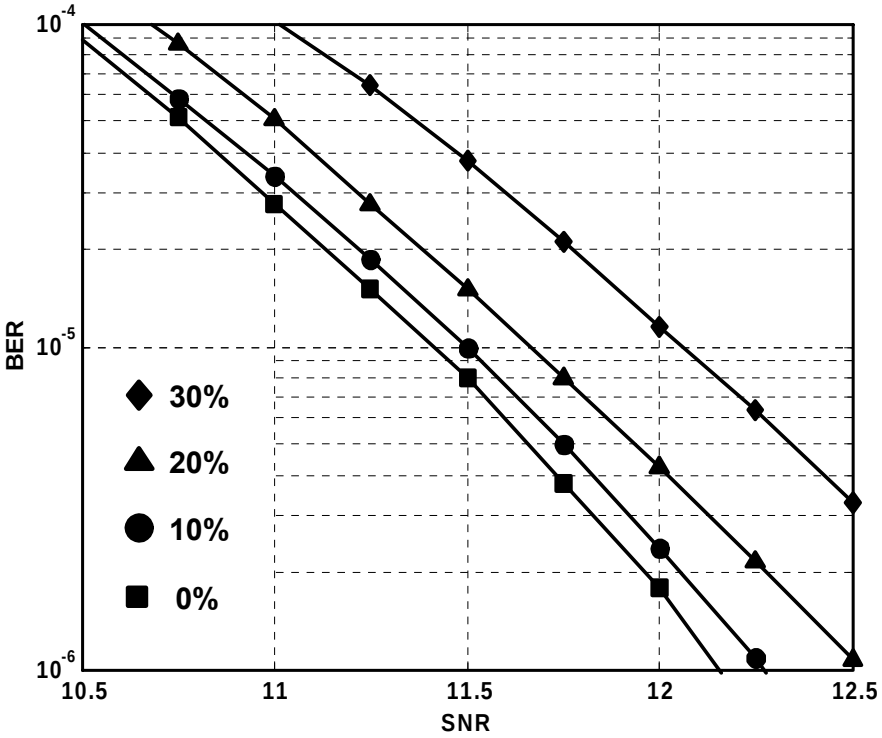


Fig. 5.27 Wi-Fi 11Mb/s CCK BER performance for different I/Q gain mismatches

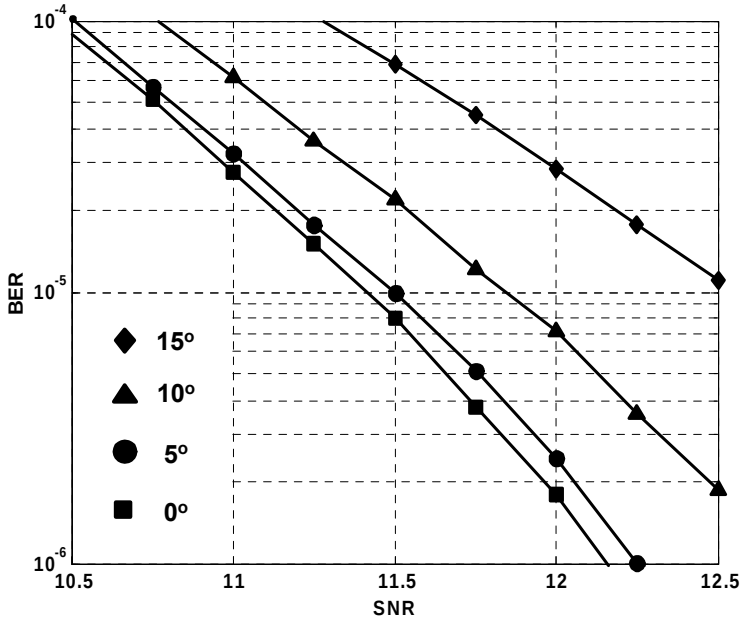


Fig. 5.28 Wi-Fi 11Mb/s CCK BER performance for different I/Q phase mismatches

5.3.7. Even-order Distortion

Typical RF receivers, with non-zero IF frequency, are susceptible to only odd-order intermodulation effects because even order distortion results in harmonics outside the band of interest. In direct conversion, on the other hand, even-order distortion becomes problematic. Suppose there are two strong interferers $A_1\cos(\omega_1 t)$ and $A_2\cos(\omega_2 t)$, close to the channel of interest placed at ω_{LO}, experience a non-linearity such as $y(t)=\alpha_1 x(t)+\alpha_2 x^2(t)$ in the LNA. Then $y(t)$ contains a 2nd order intermodulation term (IM2) term: $\alpha_2 A_1 A_2\cos(\omega_1-\omega_2)t$ indicating that two high-frequency interferers generate a low-frequency beat in the presence of even-order distortion. Upon multiplication of $\cos\omega_{LO}t$ in an ideal mixer, such a term is translated to high frequencies and hence becomes unimportant. In reality, however, mixers exhibit a finite direct feedthrough from the RF to the IF output. This feedthrough results from asymmetry in the circuit which is caused by mismatches between transistors or the deviation of the LO duty cycle from 50%. Thereby, producing an output signal such as $v_{RF}(t)(a+A\cos\omega_{LO}t)$. Thus a fraction of $v_{RF}(t)$ appears at the output with no frequency translation. If $(\omega_1-\omega_2)$ is small enough, the IM2 term appears inside the signal band and cannot be rejected by the filter.

5.3.8. VCO Frequency Pulling

An interesting phenomenon may happen when a large signal is injected into the LO. If the injected signal is close to the VCO frequency, the VCO may shift towards the injected signal frequency and eventually "lock" to that frequency. Called "injection locking," this effect is described in [45, 46]. In transceiver environment, various sources can introduce oscillator pulling. For example, the power amplifier (PA) output may

couple to the local oscillator. Another example of injection pulling arises in the receiver path when the desired signal is accompanied by a large interferer as illustrated in Fig. 5.29. If the interferer frequency is close to the LO frequency, coupling through the mixer may pull ω_{LO} toward ω_{int}. This problem is rather serious in Bluetooth since as the interferer can be as large as -27dBm, and it will be amplified to -12dBm after the LNA.

To avoid LO pulling, the VCO is run at double the RF frequency, then a divide-by-two is used to obtain the desired LO frequency. In this case, the VCO frequency is far apart from the PA and the interferer frequencies.

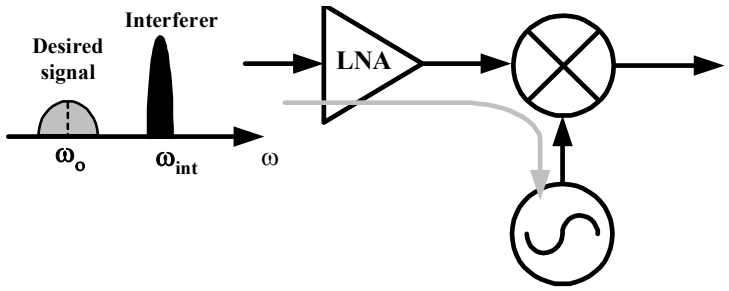


Fig. 5.29 Injection pulling due to large interferer

5.3.9. Combined Effects

BER performances of the Bluetooth and Wi-Fi demodulators have been simulated with all parameters and non-idealities mentioned in the previous subsections. The values of the parameters and non-idealities used in the simulations are listed in table 5.1 for both standards. Fig. 5.30 shows the BER performance of the Bluetooth demodulator in the ideal case and when all effects of the values listed in table 5.1 are included. Note that the required SNR_{min} becomes 13.6dB. Degradation due to the combined effects of the non-idealities is about 1.4dB. Degradation due the combined non-idealities effects in Wi-Fi mode is depicted in Fig. 5.31. The required SNR_{min} is 13.2dB, about 1.75dB above the

ideal case. These SNR_{min} values will be taken as reference in the determination of the required dual mode receiver parameters in the next section, and therefore, all non-idealities are taken into account in the estimation of the receiver performance.

Table 5.1 Receiver parameters and non-idealities used in the BER simulations

	Bluetooth	Wi-Fi
DC offset	10%	5%
HPF cutoff	1kHz	5kHz
5^{th} order Butterworth LPF cutoff	600kHz	6MHz
ADC quantization	5bits	6bits
ADC sampling rate	10MHz	44MHz
I/Q gain mismatch	10%	10%
I/Q phase mismatch	5°	5°

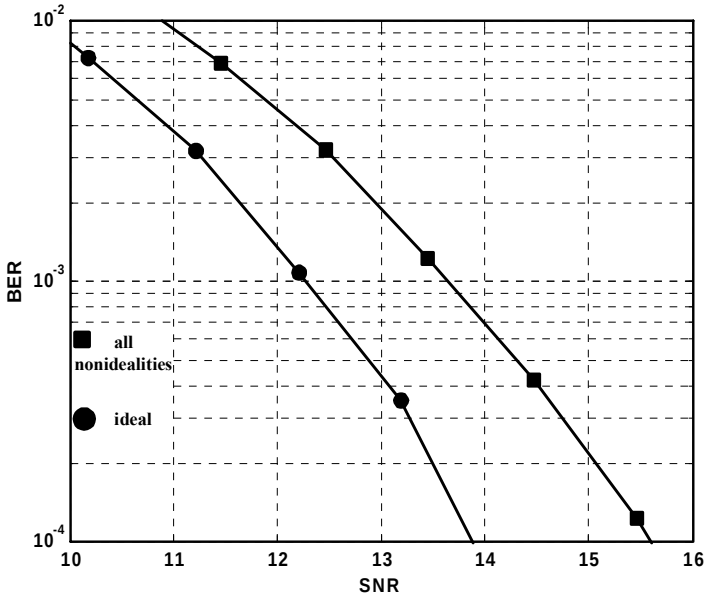


Fig. 5.30 Effects of combined receiver non-idealities on Bluetooth BER performance

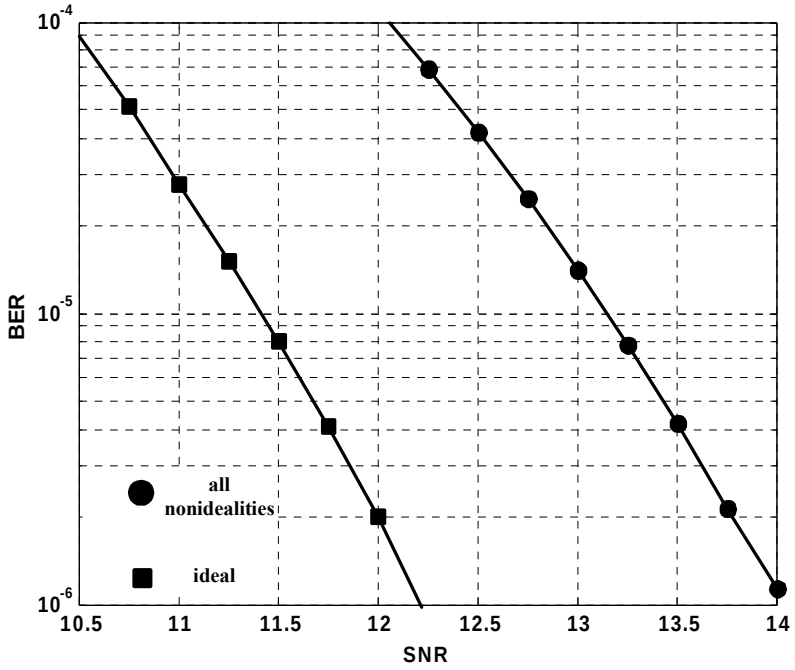


Fig. 5.31 Effects of combined receiver non-idealities on Wi-Fi BER performance

5.4. From Standard to Block Specifications

At this point, we have enough information about the two standards to start translating standard specifications into receiver system specifications, and then to individual building blocks specifications. Fig. 5.32 shows the flowchart of the entire receiver system design process.

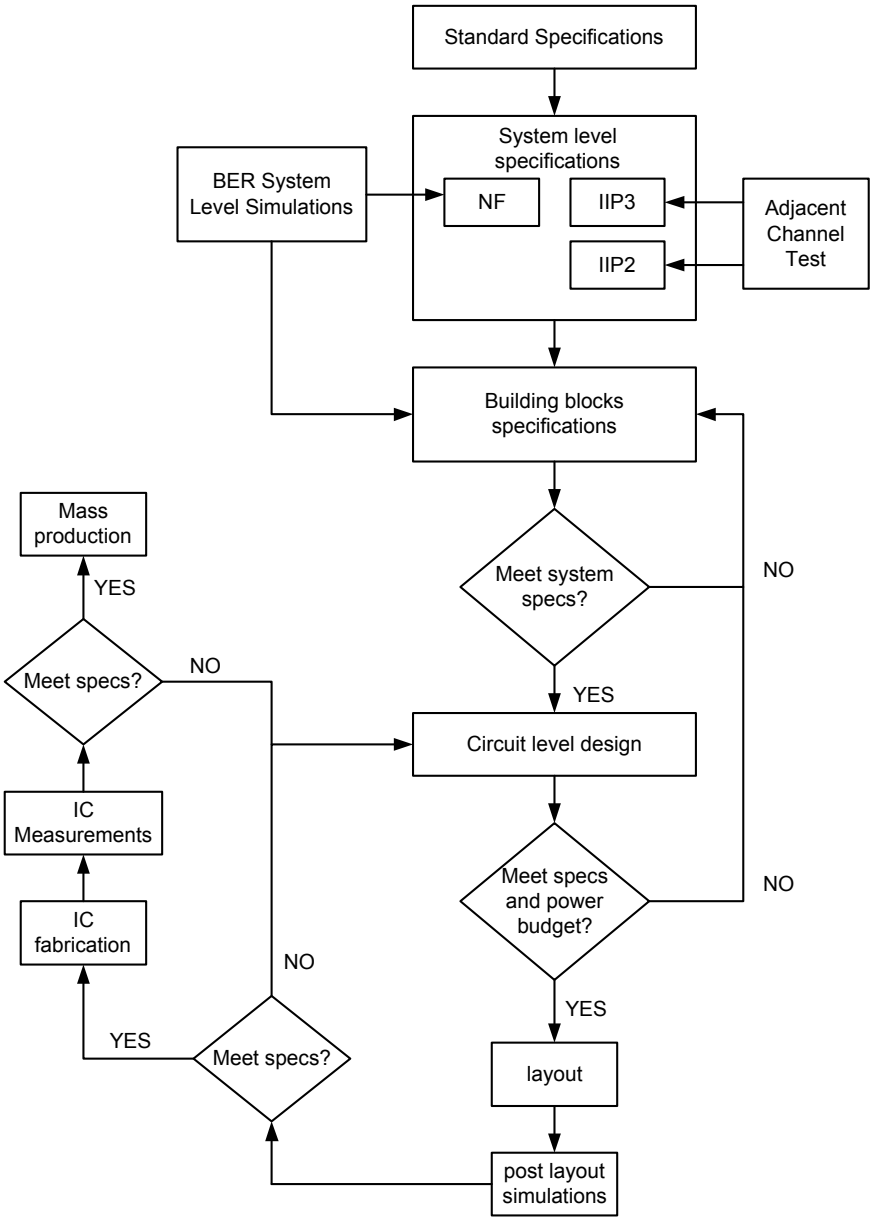


Fig. 5.32 Flowchart of the system design process

5.4.1. From Standard to Receiver Specifications

5.4.1.1. Standard receiver tests

In wireless receivers' standards, a set of receiver tests are specified to verify if the receiver complies with the standard. Among these tests, the most relevant are: sensitivity test, adjacent channel test, and intermodulation test.

Sensitivity test: Receiver sensitivity is defined as the minimum input signal power that is required to achieve a certain specified at the receiver output BER. In the sensitivity test, only the desired signal is applied to the receiver input at the required sensitivity level. The signal BER measured at the receiver output must be better than the specified BER. This test is used to derive the receiver required NF.

Adjacent channel test: In this test, the desired signal is applied to the receiver at a level, usually few decibels higher than the specified sensitivity level. In addition to the desired signal, another signal is applied at the adjacent channel frequency, at a power usually higher than the desired signal. Under these conditions, the BER measured at the receiver output must be better than the BER specified by the standard. This test is used to derive the receiver IIP2 and IIP3.

Intermodulation (IM) test: In this test, two high-power single (or modulated) tones are applied to the receiver input in presence of the desired signal. The BER measured at the receiver output must be better than the specified BER. Usually, 3rd order IM test (IM3) is specified. In this case, the two tones are applied at frequencies $f_1 = f_0 + \Delta f$ and $f_2 = f_0 + 2\Delta f$, where f_0 is the frequency of the desired channel and Δf is the two tones frequency separation. This test is usually used to determine the required receiver IIP3.

Maximum signal test: This test determines the maximum signal that can be applied to the receiver while still meeting the BER requirement. The difference between the maximum signal level and sensitivity level sets the receiver dynamic range. To have a well-defined signal level at the demodulator at the end of the receiver chain, the receiver gain has to be changed according to the signal level. Thus, sensitivity and maximum signal levels determine the maximum and minimum gain of the receiver.

5.4.1.2. Obtaining receiver key specifications

The key specifications of a direct-conversion receiver are the NF, IIP3, IIP2. These specifications are derived from the standard tests described above.

5.4.1.2.1. Noise figure (NF)

The noise floor of the receiver is expressed as:

$$\text{Noise floor} = 10\log(\text{KT}) + 10\log(BW) + NF = -174 + 10\log(BW) + NF \qquad (5.3)$$

where K is Boltzmann constant, T is the absolute temperature (290°K is usually used), NF is the receiver noise figure, and BW is the channel selection filter bandwidth. To achieve the specified BER, certain SNR_{min} must be achieved. The SNR-BER curve is obtained from baseband system level simulations as described in section 5.4.1. The receiver sensitivity can be expressed in terms of SNR_{min} and noise floor as follows:

$$\text{Sensitivity} = SNR_{\text{min}} + \text{noise floor} + \text{margin} \qquad (5.4)$$

Several dBs of margin are taken to account for possible degradation due to process variations, mismatches, or other effects that are unaccounted for in the above equation. Therefore, the required receiver NF can be expressed as:

$$NF = \text{Sensitivity} - SNR_{\text{min}} - (-174 + 10\log(BW)) - \text{margin} \qquad (5.5)$$

5.4.1.2.2. Third order intercept point (IIP3)

The IIP3 is derived from the two tone test. We can roughly state that the IM3 product must be less than the desired signal by the required SNR_{min}. In other words, we are assuming that the effect of the IM3 product is similar to the noise effect. Although this is a rough approximation, it gives a good idea about the required system IIP3 without the

need to run the actual two tone simulation. Such a simulation can take very long time because of the two higher frequency tones used.

The input referred IM3 product can be written as:

$$P_{IIM3} = 3P_{\text{int}} - 2IIP3 < P_{sig} - SNR_{\min} \tag{5.6}$$

Where P_{int} and P_{sig} are the interferer and signal levels when the IM test is conducted.

Therefore, the minimum receiver IIP3 is given by:

$$IIP3 = \frac{1}{2}(3P_{\text{int}} - P_{sig} + SNR_{\min}) + \text{margin} \tag{5.7}$$

Again, several dBs of margin are added to account for implementation loss.

5.4.1.2.3. Second order intercept point (IIP2)

Similar to the IIP3 test, this test is carried out using two nearby interferers at f_1 and $f_2 = f_1 + \Delta f$. Due to the even-order distortion of the receiver, a low-frequency beat at $f_2 - f_1 = \Delta f$ appears at the receiver output. Such a beat signal has to be less than the desired signal by the $SNR_{\min}$ value, therefore:

$$P_{IIM2} = 2P_{\text{int}} - IIP2 < P_{sig} - SNR_{\min} \tag{5.8}$$

Thus, the minimum receiver IIP2 is given by:

$$IIP2 = 2P_{\text{int}} - P_{sig} + SNR_{\min} + \text{margin} \tag{5.9}$$

5.4.1.3. Bluetooth receiver specifications

NF: The required Bluetooth receiver sensitivity is -70dBm. Baseband MATLAB simulations showed that, to achieve 10^{-3} BER, an $SNR_{\min}$ of 13.6dB is required. For 600kHz channel selection filter bandwidth, and using 6dB of margin, the required system NF is calculated by substituting these values in equation (5.5):

$$NF = \text{-}70 - 13.6 - (-174 + 10\log(600 \times 10^3)) - 6 = 26.6dB \quad (5.10)$$

This is a very relaxed number for the system NF. Many implementations [44, 47-49] have achieved sensitivities better than -80dBm. Since the required Wi-Fi NF is much tougher, it is reasonable to shoot for about -84dBm sensitivity for Bluetooth. This target sensitivity is 14dB above the specified -70dBm. Therefore, the corresponding receiver NF is 14dB less than the value obtained in equation (5.10). This gives a target NF of 12.6dB.

IIP3: In the intermodulation test, the receiver has to meet the 10^{-3} BER when the following signals are applied:

- The wanted signal at frequency f_o with a power level -64dBm.
- A static sine wave signal at f_1 with a power level of –39dBm.
- A Bluetooth modulated signal at f_2 with a power level of -39dBm.

Such that $f_o = 2f_1 - f_2$ (f_1 is closer to the signal than f_2) and $|f_2 - f_1| = n \times 1MHz$, where n can be 3, 4, or 5. The system must fulfill one of these alternatives.

Substituting these values in equation (5.7), using 6dB margin:

$$IIP3 = \frac{1}{2}(3(-39) - (-64) + 13.6) + 6 = -13.7dBm \quad (5.11)$$

IIP2: There is no specified IIP2 test in the Bluetooth standard. However, the IIP2 can be derived from the adjacent channel test or the intermodulation test, whichever requires higher IIP2. The largest specified interferer is -27dBm Bluetooth modulated signal at offset frequency ≥ 3MHz, where the signal is at $-67dBm$. Since the interferer is frequency modulated (no amplitude modulation), 2nd order distortion in the receiver will result in dc offset. Unfortunately, AC coupling does not reject this dc offset because the

LNA input is grounded during the dc offset estimation (to avoid corrupting dc information by the incoming signals). However, this offset can be rejected during the synch word before receiving the actual data. Therefore, the dc offset does not have to be much smaller than. This offset has to be lower than the signal by SNR_{min} = 13.6dB. This can be viewed as if we have two very close interferers, with power $-27-3=-30dBm$ each. Therefore from equation (5.9):

$$IIP2 = 2(-30)-(-67)+13.6+6 = 26.6dBm \tag{5.12}$$

5.4.1.4.Wi-Fi receiver specifications

NF: The sensitivity level at 11Mbit/s data rate is -76dBm with a frame error rate (FER) of 0.08. The frame length is 1024 octets (or equivalently 8×1024 = 8092bits). So this FER corresponds to a BER of 0.08/8092 ≈ 10^{-5}. Based on *System View* Baseband simulations, the SNR_{min} to achieve 10^{-5} BER is 13.2dB. If the channel selection filter bandwidth is 6MHz, and using 6dB of margin, the required system NF from equation (5.5) is:

$$NF = \text{-}76-13.2-(-174+10\log(6\times10^{6}))-6 = 11dB \tag{5.13}$$

In this design, our target sensitivity is -80dBm, 4dB above the Wi-Fi sensitivity specification. Therefore, the required NF is 11-4 = 7dB.

IIP3: There is no standalone IIP3 test in Wi-Fi standard. However, the required IIP3 can be estimated from the adjacent channel test. In this test, an 11Mbit/s Wi-Fi signal is applied at 25MHz away from the desired signal. The desired and interference signal levels are -70dBm and -35dBm, respectively. Since the interferer is a wideband signal, direct application of equation (5.7) is not accurate. A MATLAB program is developed (appendix C) to compare the intermodulation products of the CCK signal at 25MHz away

from the signal with the intermodulation product of two tones at 16.7 and 33.4MHz at the same power level. If the 3rd harmonic product is passed through a 5th order 6MHz filter, the CCK IM3 output will be 37dB less than the two tones IM3 product. Therefore, in order to calculate the required system IIP3, equation (5.6) will be modified as follows:

$$3P_{\text{int}} - 2IIP3 - 37 < P_{sig} - SNR \tag{5.14}$$

Then the minimum IIP3 in Wi-Fi mode is:

$$\begin{aligned} IIP3 &= \frac{1}{2}(3P_{\text{int}} - 37 - P_{sig} + SNR_{\min}) + \text{margin} \\ &= \frac{1}{2}(3\times(-35) - 37 + 70 + 13.2) + 6 = -23.4dBm \end{aligned} \tag{5.15}$$

It is worth mentioning that the IM3 product of the CCK signal is obtained by convolving the magnitude of the CCK spectrum with itself three times. This implies that the obtained IM3 product is actually a worst-case number. In other words, the required IIP3 calculated in (5.15) is rather a conservative value.

IIP2: Again, there is no IIP2 test in Wi-Fi standard. Therefore, we will follow the same approach used in IIP3 calculations to calculate the IIP2 using the adjacent channel test. A MATLAB code (appendix C) is developed to compare the IM2 product of the CCK signal with the IM2 in the standard IIP2 test using two tones at 24 and 26MHz. The CCK IM2 product at the output of a 6MHz 5th order Butterworth filter is 20dB less than the two tone IM2 product. Therefore, equation (5.8) can be modified as follows:

$$2P_{\text{int}} - IIP2 - 20 < P_{sig} - SNR \tag{5.16}$$

Then the minimum required IIP2 is estimated as:

$$IIP2 = 2P_{\text{int}} - 20 - P_{sig} + SNR_{\min} + \text{margin} = 2(-35) - 20 - (-70) + 13.2 + 6 = -0.8dBm \tag{5.17}$$

A summary of required receiver specifications for both modes is listed in table 5.2.

Table 5.2 Summary of receiver specifications

Specifications	Value	
	Wi-Fi	Bluetooth
NF (dB)	7	12.6
IIP3 (dBm)	-23.4	-14.4
IIP2 (dBm)	-0.8	26.6

5.4.2. Determination of Receiver Building Blocks Specifications

From the receiver system specifications, the specifications of the individual building blocks can be obtained. Conversion from system to blocks specifications is rather an iterative process. First we start with a reasonable distribution of specifications based on previous experience and/or some intuition. Then the overall system specifications are calculated to see if the system specifications are met. The distribution of blocks specifications is subject to the restriction of sharing the same RF front-end (LNA, mixer, and PLL) from both standards to save silicon area. In other words, the specifications of these blocks will be set by the toughest requirement of both standards. Although a set of key receiver specifications (NF, IIP3, and IIP2) were obtained in the previous section, we still need to look at the standard tests, from time to time, to make sure that the receiver is operating properly under all conditions. A MATLAB code (appendix D) is developed to calculate the system parameters from the blocks specifications. This program is used to check if the systems specifications are met for a set of blocks specifications.

5.4.2.1. Gain distribution

The first step to determine the specifications of the receiver blocks is to determine the gain distribution. In order to relax the NF of the baseband blocks (filter, VGA, and ADC), RF front-end gain should be maximized. The maximum RF front gain is determined by the adjacent channel test. In Bluetooth mode, the largest adjacent channel

interferer is -27dBm. If the maximum signal at the output of the mixer is 4dBm ($0.5V_p$), the total RF front-end (LNA+mixer) gain is 31dB. In the Wi-Fi case, the adjacent channel power level is -35dBm, which is lower than Bluetooth interferer, and therefore, will not saturate the mixer output. Since the interferer level is largest at the filter input, filter linearity is very critical. An odd order filter is preferred because of the availability of a passive pole that can be placed at the filter input to attenuate the interferer, and therefore, improve the out-of-band linearity of the filter. A 5^{th} order Butterworth filter is implemented. To further relax the NF of the filter and the VGA, a 6dB gain is implemented at the filter's first biquad (after the passive pole). The maximum signal swing at the ADC input is $2V_{p\text{-}p}$. To allow for some margin, the total receiver gain is adjusted such that the signal level at the ADC input is $1V_{p\text{-}p}$ (4dBm). To achieve Wi-Fi sensitivity of -85dBm, the overall maximum receiver gain is 89dB. Therefore, the maximum VGA gain is $89-31-6=52dB$. In Wi-Fi mode, the maximum signal is -4dBm for the 2MHz data rate. This level is too high for the RF front-end and will saturate the LNA. For such high signal levels, LNA is bypassed and the signal is passed through an attenuator (low gain mode). To avoid saturating the filter and VGA, the attenuator gain is chosen such that the signal level at the output of the filter is below 4dBm. If the RF front-end low gain is 1dB (30dB drop from the high gain mode), the signal level at the output of the filter is 3dBm, below the maximum level. To get a 4dBm signal level at the ADC input, the minimum VGA gain is 1dB. According to typical values, the LNA and mixer gains are chosen to be 15 and 16dB, respectively. Therefore, to have a 30dB gain drop from high gain to low gain, the attenuator provides 15dB of attenuation. Table 5.3 shows the gain distribution of the receiver blocks. To cover for all possible errors in

the gains of the blocks, especially the RF blocks, the VGA gain is designed for a gain range of 0 to 62dB.

In Bluetooth mode, due to the short preamble time, the overall receiver gain has to be adjusted using the minimum number of steps. Changing the front-end gain between high and low gain modes is unavoidable due to the large maximum signal level in Bluetooth (-20dBm). If we shoot for sensitivity and maximum signal levels of -90dBm and -10dBm, signal dynamic range at the receiver input is 80dB. Using 30dB gain step in the front-end, the signal level at the VGA input can vary by about 50dB. If the VGA gain is fixed in Bluetooth mode to minimize receiver gain settling time, signal dynamic range at the ADC input will be 50dB, requiring more than a 13bits ADC to cover such dynamic range. Therefore, only one large gain step is used in the VGA to reduce the signal dynamic range at the ADC input. At maximum signal input (-4dBm in Wi-Fi mode), the LNA is bypassed by the attenuator and the VGA gain is set to 0dB. At the sensitivity level, -90dBm, the front-end is set to the high gain mode and the VGA gain is set at 24dB at which the signal level at the ADC input is -29dBm. Thus, in Bluetooth mode, the signal dynamic range at the ADC input is 80-30-24=26dB, requiring about 10bits of ADC resolution which is a reasonable value for the Bluetooth data rate.

Table 5.3 Receiver gain distribution

	LNA/attenuator	Mixer	Filter	VGA (Wi-Fi/BT)
Maximum gain (dB)	15	16	6	62/24
Minimum gain (dB)	-15	16	6	0

5.4.2.2. NF distribution

Starting with the gain distribution in table 5.3, we can distribute the NF among the receiver block such that the required overall system NF is met. The overall system NF is then calculated using the Friis equation [50] for cascaded stages:

$$NF_{tot} = 1 + (NF_1 - 1) + \frac{NF_2 - 1}{A_1^2} + \frac{NF_2 - 1}{A_1^2 A_2^2} + \cdots + \frac{NF_m - 1}{A_1^2 A_2^2 \cdots A_{m-1}^2} \tag{5.18}$$

Where NF_i and A_i are the NF and gain of the i^{th} stage, where the LNA is the first stage. The Friis equation indicates that the noise contributed by each stage decreases as the gain of the preceding stages increases, implying that the first few stages in a cascaded are the most critical. In the case of our particular receiver, since the RF front-end gain is limited by the adjacent channel test, NFs of LNA, mixer, and filter dominate the overall system NF. Table 5.4 lists the NF of the receiver building blocks. Percentage contributions to the overall system NF in Bluetooth and Wi-Fi modes are shown in the pie diagrams in Fig. 5.33 and 5.34, respectively.

Table 5.4 NF, IIP3, and IIP2 distribution in Bluetooth/Wi-Fi mode

	LNA	Mixer	Filter	VGA	ADC	System
NF (dB)	3/3	20/15	36/32	30/30	57/65	9.49/6.46
IIP3 (dBm)	-8	5	23	10	10	-13.5
IIP2 (dBm)	11	48	64	31	41	26.7

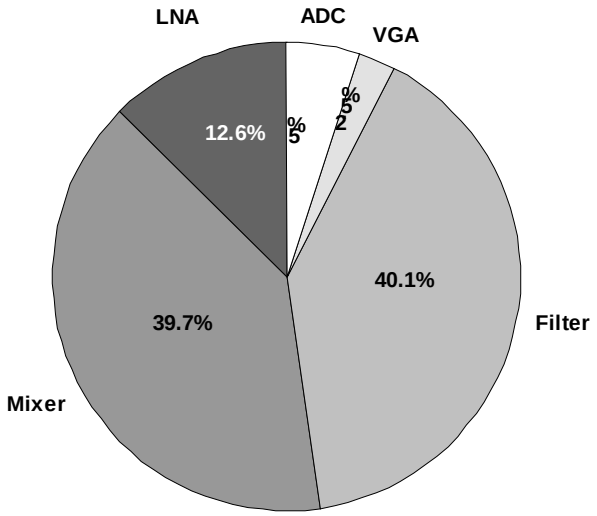


Fig. 5.33 Noise contributions from the different receiver blocks in Bluetooth mode

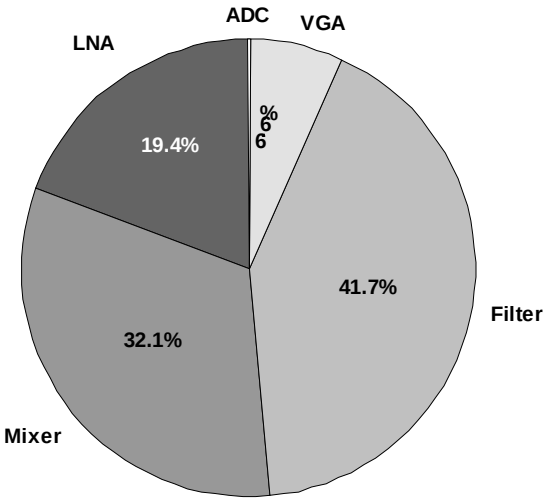


Fig. 5.34 Noise contributions from the different receiver blocks in Wi-Fi mode

5.4.2.3. IIP3 distribution

Since the interferer is filtered out by the LPF, the IIP3 of the receiver will be dominated by the LNA, mixer, and LPF. The IIP3 of the RF front end will be the same for both Bluetooth and Wi-Fi. Furthermore, it will be easier to design the filter if it has

the same IIP3 in both modes. Therefore, the IIP3 specification will be determined by the tougher of the two standards which is -14.4dBm in Bluetooth mode. The IIP3 is distributed among the LNA, mixer, and LPF as shown in table 5.4. Contribution of the three blocks to the system IIP3 is depicted in Fig. 5.35. The overall system IIP3 is calculated from the IIP3 of the individual blocks using the following equation:

$$\frac{1}{A_{IIP3_{tot}}^2} = \frac{1}{A_{IIP3_1}^2} + \frac{A_1^2}{A_{IIP3_1}^2} + \frac{A_1^2 A_2^2}{A_{IIP3_3}^2} + \cdots + \frac{A_1^2 A_2^2 \cdots A_{m-1}^2}{A_{IIP3_m}^2} \tag{5.19}$$

Where A_{IIP3_i} and A_{i_i} is the input referred IIP3 (in volts) and the linear voltage gain of the i^{th} block.

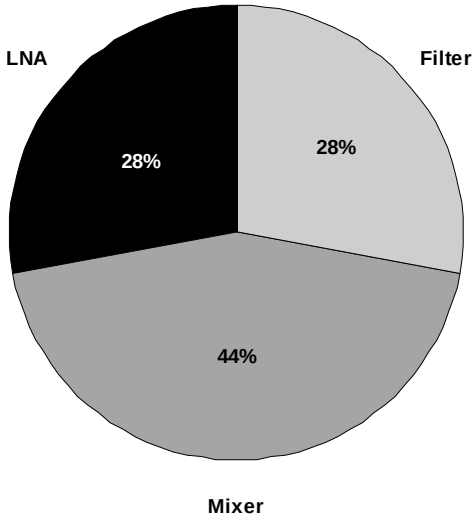


Fig. 5.35 IIP3 contributions from receiver blocks in both modes

5.4.2.4. IIP2 distribution

Similar to the IIP3, the system IIP2 is typically dominated by the mixer and the filter. Ideally, all low-frequency components created by the even-order distortion on the LNA

gets upconverted by the mixer, and consequently filtered out by the LPF. However, If a single-ended mixer is used, RF-IF isolation is poor and the even-order harmonic at the output of the LNA propagate through the mixer, corrupting the downconverted signal spectrum. Therefore, the IIP2 of the LNA contributes to the system IIP2 if a single-ended mixer is used. For differential mixers, RF-IF isolation is better (typically around 30dB) and it depends on matching between transistors. Differential double balanced mixer is used in this design receiver to have good RF-IF and LO-RF isolation. LO-RF isolation is important to alleviate the self-mixing problem discussed in section 5.4.2. The IIP2 is distributed as shown in table 5.4. Contributions of the receiver blocks to the system IIP2 are depicted in Fig. 5.36. The following equation is used to calculate the system IIP2 based on the IIP2 of the individual blocks.

$$\frac{1}{A_{IIP2_{tot}}} = \frac{1}{A_{IIP2_1}} + \frac{A_1}{A_{IIP2_2}} + \frac{A_1 A_2}{A_{IIP2_3}} + \cdots + \frac{A_1 A_2 \cdots A_{m-1}}{A_{IIP2_m}} \tag{5.20}$$

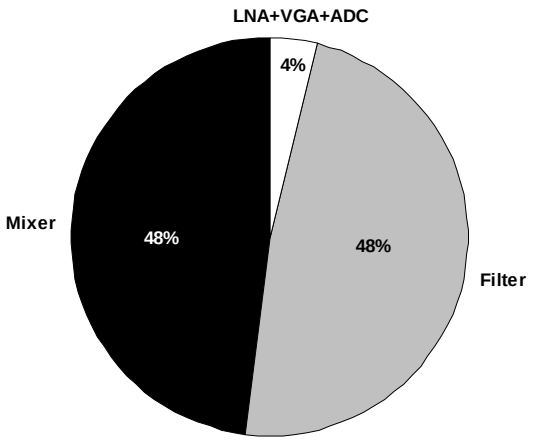


Fig. 5.36 IIP2 contributions from receiver blocks in both modes

5.4.2.5. LO phase noise

Ideally, the output of the LO is a single tone signal at the frequency of the desired signal (in DCR architecture). However, due to thermal and 1/f noise of the transistors/resistors used in the PLL, amplitude and phase noise components appear at the output of the LO. Amplitude noise is not critical because of the hard switching of the mixer transistors. Phase noise, on the other hand, is very critical especially in the presence of strong out-of-band interferers. This is due to the phase noise of the LO modulating the carrier of the strong signal. The carrier is spread in frequency by the phase noise modulation, which results in a power spectral density that is proportional to the LO phase noise. This effect is called *reciprocal mixing*, and is illustrated in Fig. 5.37. Phase noise is usually specified at a specific offset frequency from the LO carrier frequency. For example phase noise of -100dBc/Hz[4] at 1MHz offset means that the total noise power in 1Hz bandwidth at 1MHz offset from the LO center frequency is 120dB below the LO carrier power.

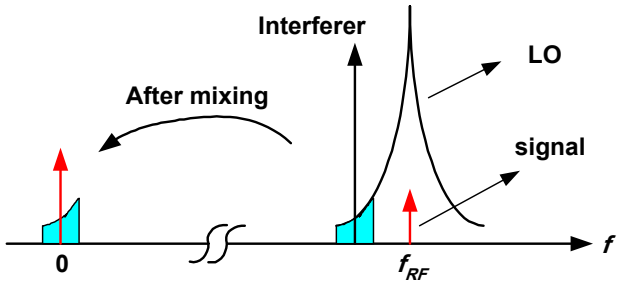


Fig. 5.37 LO phase noise effect

The LO phase noise requirement is derived from the adjacent channel test. In the case of Bluetooth, the derivation of LO phase noise is quite straightforward because the

[4] LO phase noise is usually specified in dBc/Hz which represents the noise power in 1Hz bandwidth at the specified offset frequency relative to the LO fundamental output power.

interferer is narrow band Bluetooth modulated signal which we can approximate as a single tone signal. If the adjacent channel power is P_{int}, at Δf offset frequency, and the signal power is P_{sig}, then the signal to phase noise ratio (SPNR) is expressed as:

$$SPNR = P_{sig} - (P_{\text{int}} + PN + 10\log(2BW)) = C/I - PN - 10\log(2BW) \qquad (5.21)$$

Where C/I is the carrier to interference ratio and BW is the channel selection filter bandwidth. BW is multiplied by 2 to account for both sidebands around the offset frequency. If the signal is x dB above sensitivity (x is specified in the standard), then the signal to thermal noise ratio is $SNR_{\min} + x$ (assuming same receiver gain distribution as in the sensitivity test). Therefore, the overall SNR at the output of the receiver is given by:

$$10^{SNR/10} = \frac{1}{10^{-SPNR/10} + 10^{-(SNR_{\min}+x)/10}} \qquad (5.22)$$

To get the maximum LO phase noise, we set the overall SNR=SNR_{min}:

$$10^{SNR_{\min}/10} = \frac{1}{10^{-(C/I-PN-10\log(2BW))/10} + 10^{-(SNR_{\min}+x)/10}} \qquad (5.23)$$

Therefore, the required LO phase noise, with some margin, is given by:

$$PN = C/I - SNR_{\min} - 10\log(2BW) - 10\log(1-10^{-x/10}) \qquad (5.24)$$

Table 5.5 shows the different interferer levels, offset frequencies and the corresponding required LO phase noise for SNR_{min} = 13.6dB and BW = 600kHz.

Table 5.5 Required LO phase noise in Bluetooth mode

Δf (MHz)	C/I (dB)	x (dB)	PN (dBc/Hz)
1	0	10	-74.85
2	-30	10	-104.85
3	-40	3	-117.41

At the offset frequencies listed in table 5.5, the LO phase noise is proportional to *1/f*. Therefore, the phase noise value at 3MHz offset is the toughest to achieve and hence will be taken as the specification for the LO phase noise.

In Wi-Fi mode, the interferer must be considered as a wideband signal; therefore equation (5.21) can be rewritten as:

$$\begin{aligned} SPNR &= P_{sig} - P_{\text{int}} - 10\log(\int_{-BW}^{BW} CCK_n(f) * 10^{PN(f+\Delta f)/10} df) \\ &= C/I - PN_{eff} - 10\log(2BW) \end{aligned} \tag{5.25}$$

Where PN_{eff} is the LO effective phase noise for a CCK signal, $CCK_n(f)$ is the normalized (power = 1) CCK signal spectrum. Assuming 1/f dependence for phase noise, therefore:

$$PN_{eff} = 10\log(\int_{-BW}^{BW} 10^{CCK(f)/10} * \frac{K}{f+\Delta f} df) - 10\log(2BW) \tag{5.26}$$

In Wi-Fi, the interferer is at $\Delta f = 25MHz$ away at -35dBm, while the signal level is -70dBm, 6dB above sensitivity. Substituting these values in equation (5.26), the required effective phase noise is -111dBc/Hz. Simulations using MATLAB (appendix C) showed that in order to obtain an effective PN of -111dBc/Hz for CCK, the actual phase noise at 25MHz is -99.2dBc/Hz, 13.8dB above the effective value. This is equivalent to -90dBm at 3MHz offset if 1/f PN behavior is assumed, which is about 27.4dB more relaxed than the required PN in Bluetooth mode. Since this assumed case is much more stringent that the actual wideband adjacent channel, we conclude that the PN of the receiver will be determined by the Bluetooth rather than the Wi-Fi standard.

5.4.3. System Design Verification

Once all the blocks specifications are determined, we need to make sure that the overall system complies with the standards specifications by passing it through the standards tests described in section 5.5.1.1. This is done by using the MATLAB code in appendix D. In these tests, we have to make sure that the signal level at the input of output of every block is within the dynamic range of that block and that the required SNRmin is achieved at all input signal levels.

5.4.3.1. System verification in Bluetooth mode

Fig. 5.38 illustrates the signal level at the ADC input versus the RF input level. As mentioned before, the LNA is activated and the VGA gain is set to minimum at startup. In other words, the signal is initially assumed to be in the range from -57dBm to -33dBm as indicated in Fig. 5.38. Based on the measured digital signal level at the output of the ADC, a decision will be taken to bypass the LNA by the attenuator or to switch the VGA gain to maximum or to keep the current gain distribution. The gain switching strategy is depicted in Fig. 5.39 and is designed such that the signal level at the ADC input is below 4dBm and above the ADC quantization noise floor by at least 24dB. Fig. 5.40 shows the signal and noise levels along the receiver chain in the Bluetooth sensitivity test (a target sensitivity of -84dBm is assumed). The signal level at the ADC input is -23dBm, well above the ADC quantization noise floor which is about -50dBm. The SNR at the input of each block is depicted in Fig. 5.41. The adjacent channel test is shown in Fig. 5.42. Note that in this test, the interferer level is at maximum (4dBm) at the filter input. The figure shows that, as pointed out earlier, this test determines the maximum RF gain that can be employed. The filter has 6dB gain and it attenuates the interferer by 54dB (signal gain –

interferer gain = 60dB). Fig. 5.43 depicts the Bluetooth maximum signal test. The specified maximum Bluetooth is -20dBm. The LNA is bypassed by the -15dB attenuator and the VGA gain is set to minimum (0dB). The signal level at the ADC input is -13dBm, which means that even higher input RF signal powers can be tolerated in the Bluetooth mode. At the input level -33dBm when the LNA/attenuator is switched, proper operation of the receiver in either case (LNA is activated or bypassed) must be verified. Fig. 5.44 shows the signal and noise levels when the LNA is active (high gain) and VGA is at low (0dB) gain for the -33dBm signal level. When the LNA is bypassed by the attenuator, the signal level at the ADC input drops as shown in Fig. 5.45. Figs. 5.40-5.45 are obtained from the MATLAB code in section D.1.

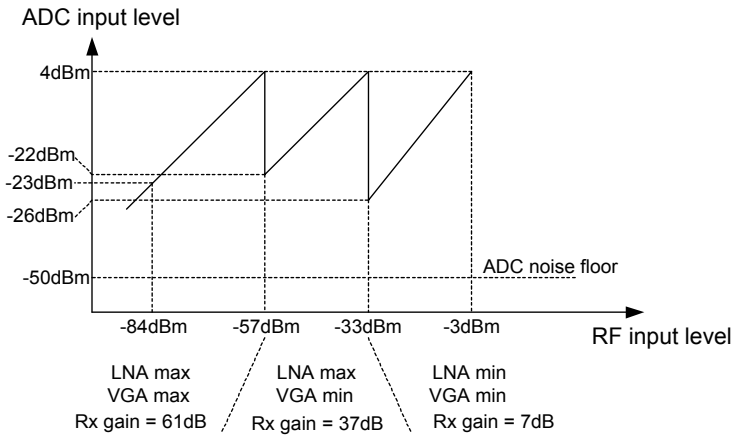


Fig. 5.38 ADC input level versus RF input power in Bluetooth mode

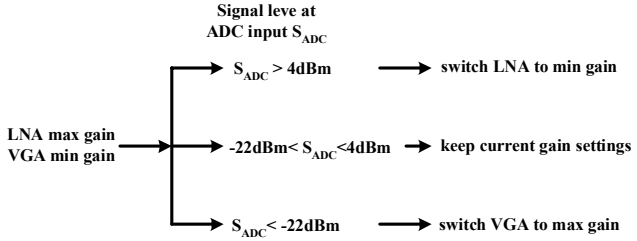


Fig. 5.39 Gain switching strategy in Bluetooth mode

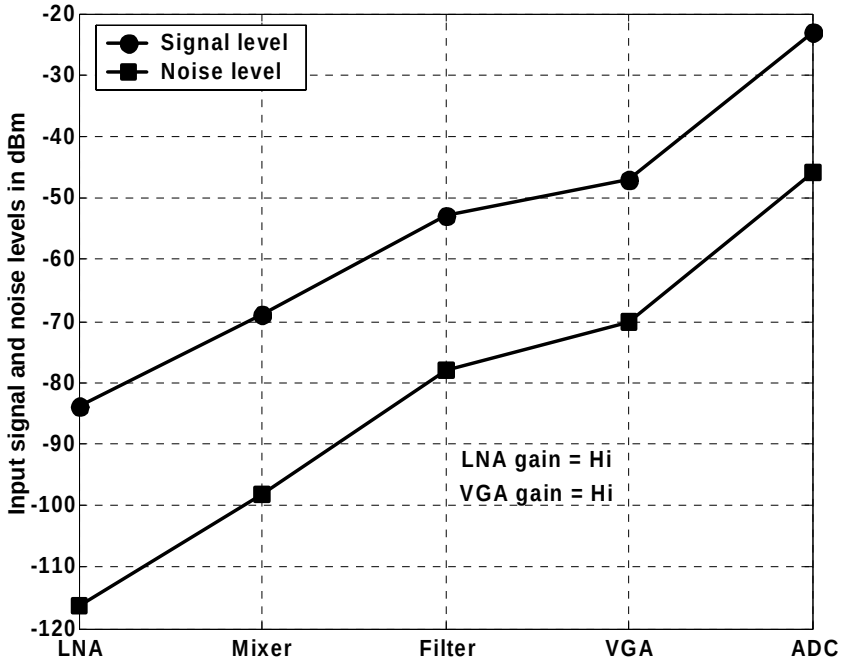


Fig. 5.40 Signal and noise levels in Bluetooth sensitivity test

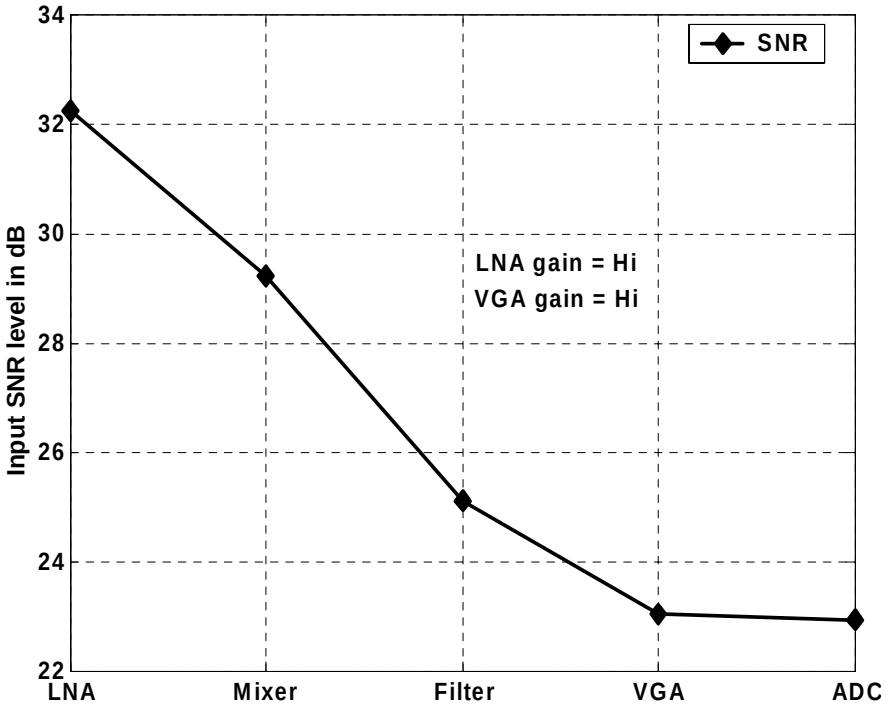


Fig. 5.41 Signal-to-noise ratio in Bluetooth sensitivity test

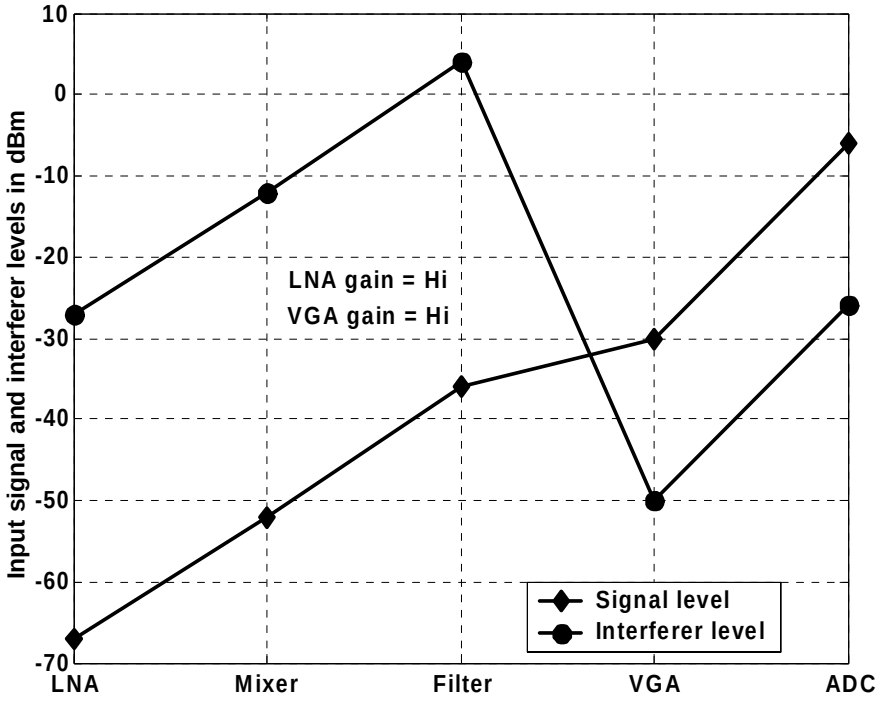


Fig. 5.42 Adjacent channel test in Bluetooth mode

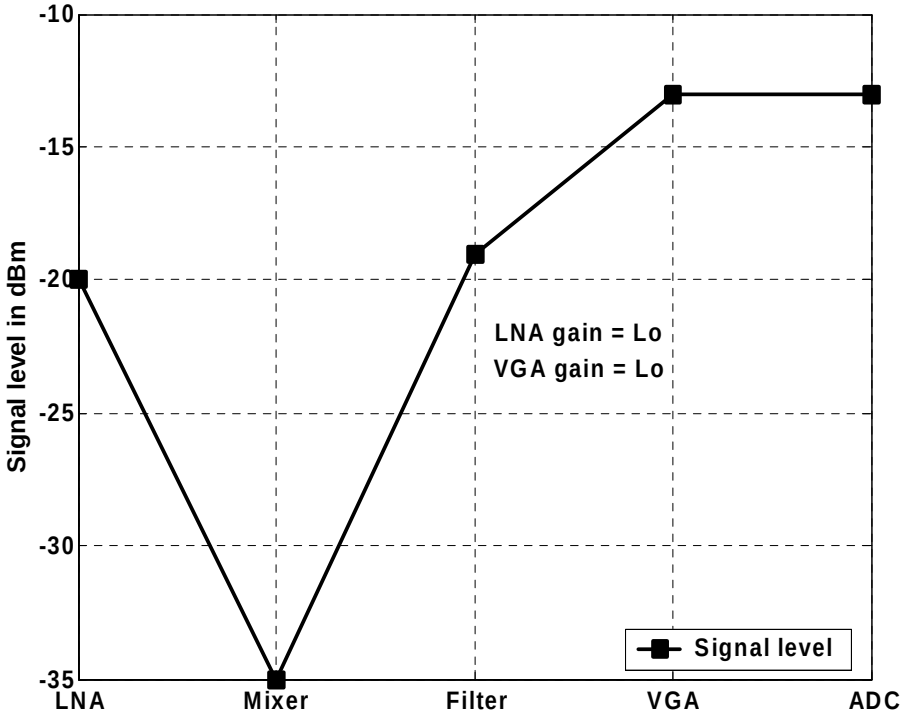


Fig. 5.43 Bluetooth maximum signal test

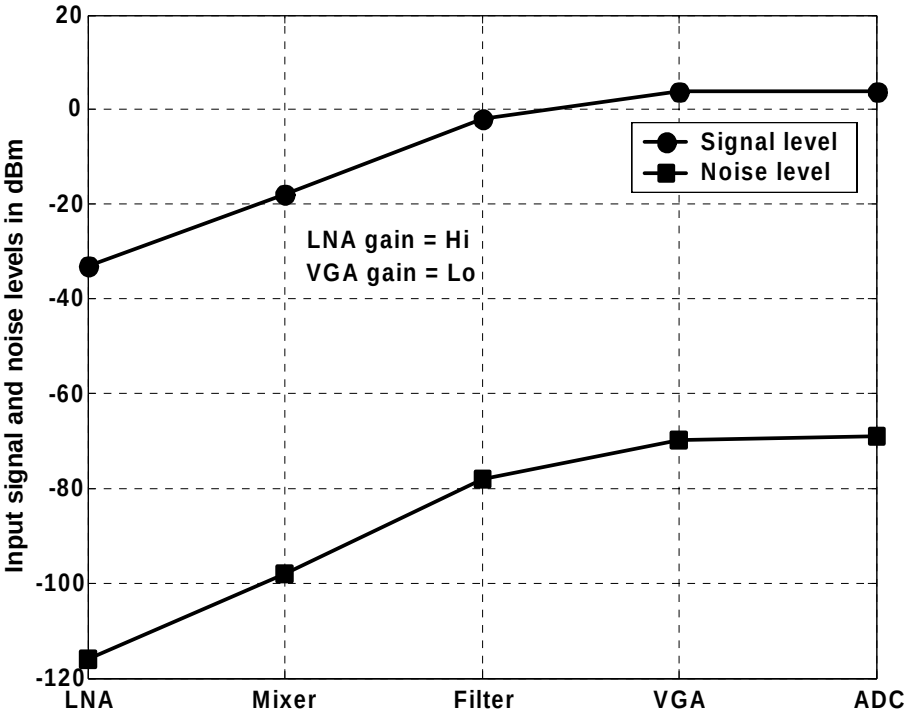


Fig. 5.44 Signal test at gain switching point (LNA gain is activated and VGA gain is low)

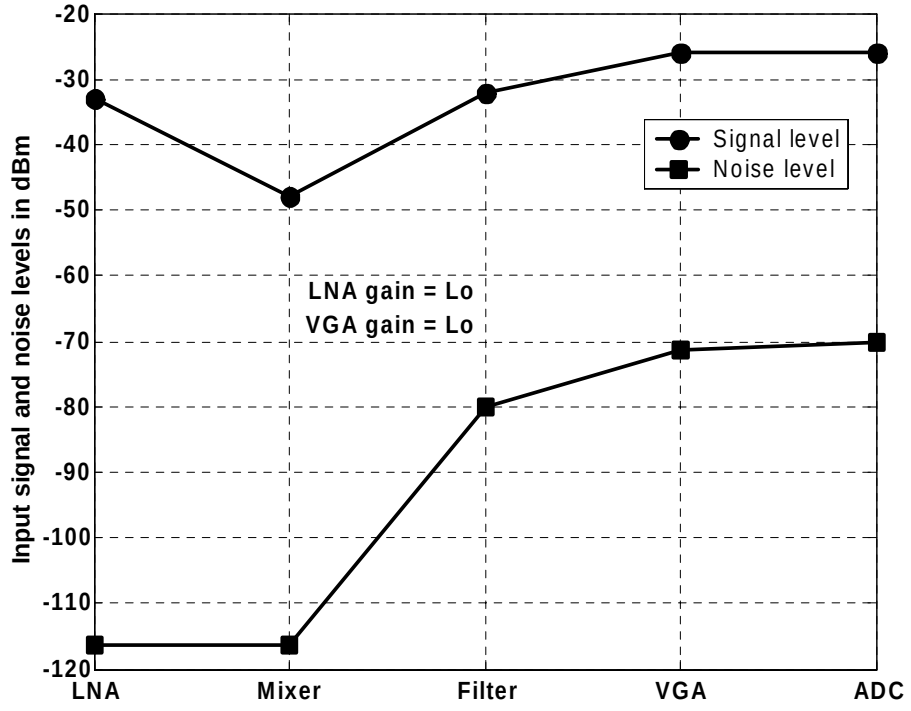


Fig. 5.45 Signal test at gain switching point (LNA is bypassed and VGA gain is low)

5.4.3.2. System verification in Wi-Fi mode

The same tests are performed in Wi-Fi mode. The signal level at the ADC input is plotted versus the input RF signal level in Fig. 5.46. Note that due to the fine VGA gain steps, the signal level at the ADC input is well controlled within ±1dBm for the entire desired signal level range. The LNA is bypassed if the RF signal level is -37dBm. The sensitivity test is performed at a target sensitivity of -80dBm. The gain switching strategy is depicted in Fig. 5.47. Fig. 5.48 shows the signal and noise levels at the inputs of the building blocks while the SNR is depicted in Fig. 5.49. The SNR at the ADC input is about 19.8dB, about 6dB above SNRmin for Wi-Fi mode. This is consistent with the 6dB margin taken in the NF calculations (equation (5.13)). The adjacent channel test is plotted in Fig. 5.50. 50dB interferer attenuation by the channel selection LPF is assumed. Since the interferer is 35dB higher than desired signal, the signal-to-interference ratio is 15dB at the ADC input. The maximum signal test is depicted in Fig. 5.51. The RF input signal level is -4dBm (for the low rate, 2Mbit/s, in 802.11 standard). LNA is bypassed by the attenuator and the VGA gain is set at minimum. The signal level at the ADC input is 3dBm which is still within the ADC dynamic range. At the signal level -37dBm, the receiver must operate properly whether the LNA or the attenuator is activated to insure receiver compliance to the Wi-Fi standard at all signal levels. Fig. 5.52 shows the signal and noise levels when the LNA is activated. The VGA gain is set to its minimum (0dB) and the signal level at the ADC is at 0dBm. Fig. 5.53 depicts the test when the LNA is bypassed by the attenuator at the same signal level (-37dBm). The SNR at the ADC input is higher than 30dB and the signal level at the ADC input is 0dBm.

Therefore, all the plots ensure that, with the block specifications determined in the previous subsection, the overall system is meeting the standards specifications. Fig. 5.48-5.53 are obtained from the MATLAB code in section D.2.

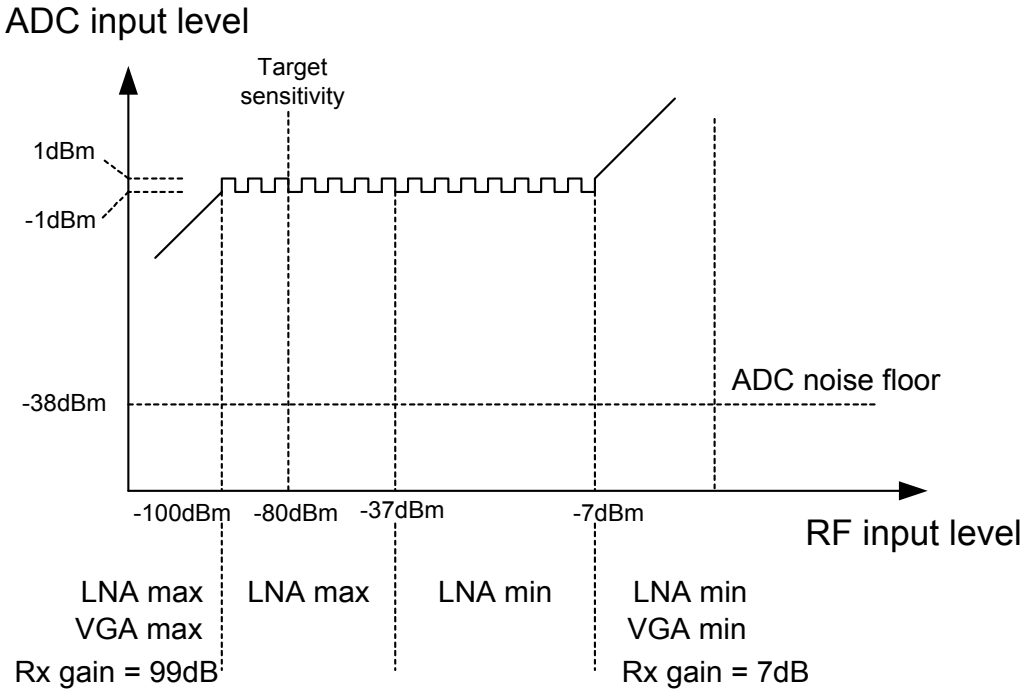


Fig. 5.46 ADC input level versus RF input power in Wi-Fi mode

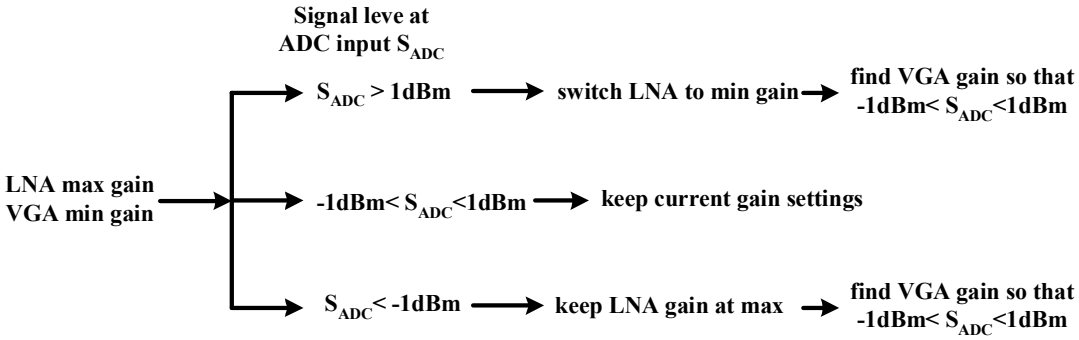


Fig. 5.47 Gain switching strategy in Wi-Fi mode

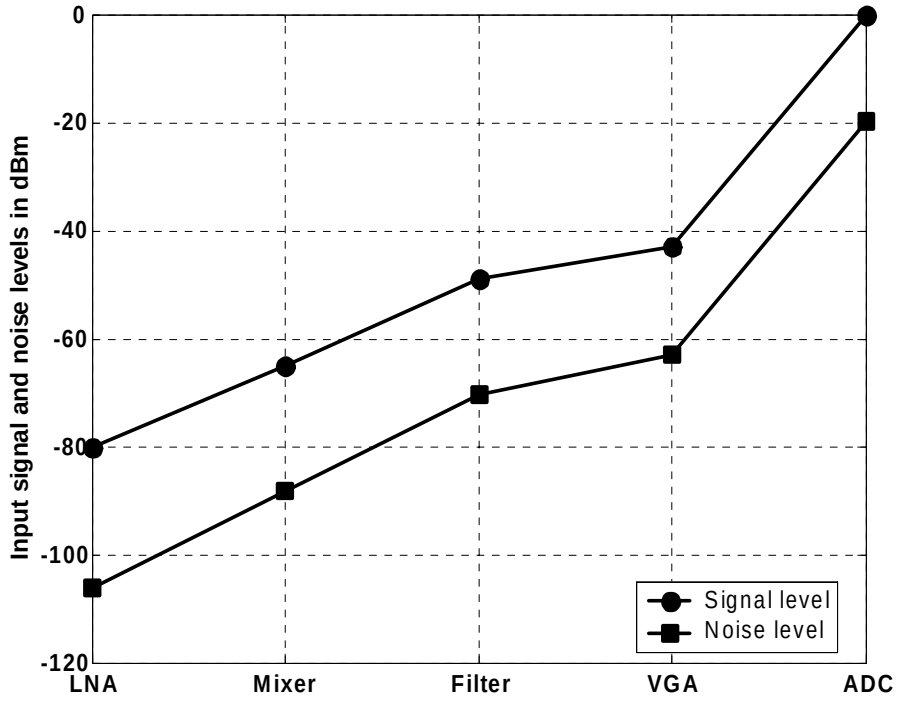


Fig. 5.48 Signal and noise levels in Wi-Fi sensitivity test

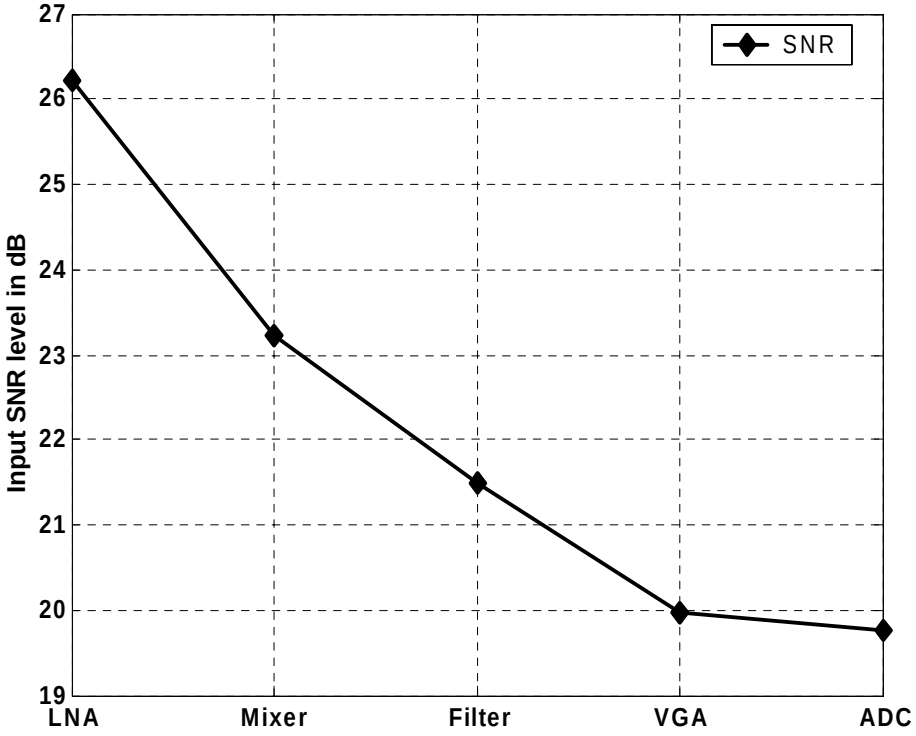


Fig. 5.49 Signal-to-noise ratio in Wi-Fi sensitivity test

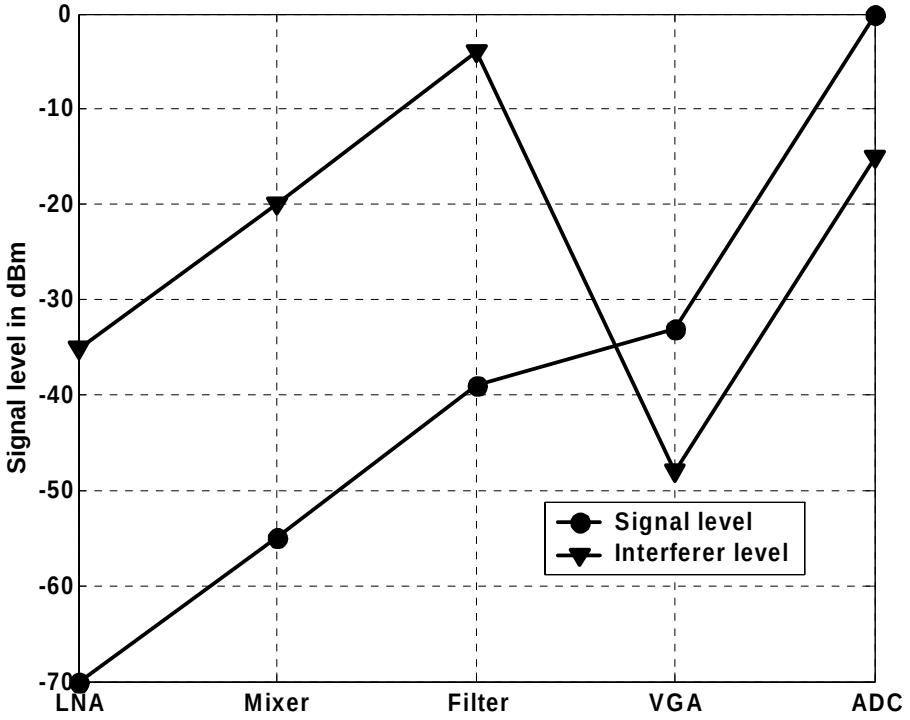


Fig. 5.50 Adjacent channel test in Wi-Fi mode

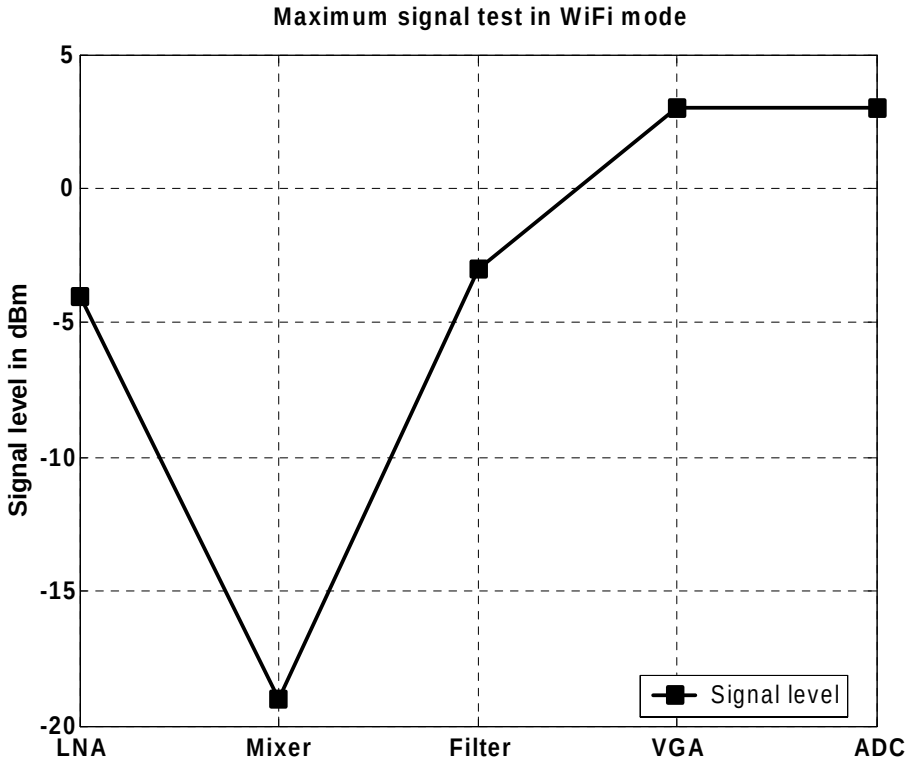


Fig. 5.51 Wi-Fi maximum signal test

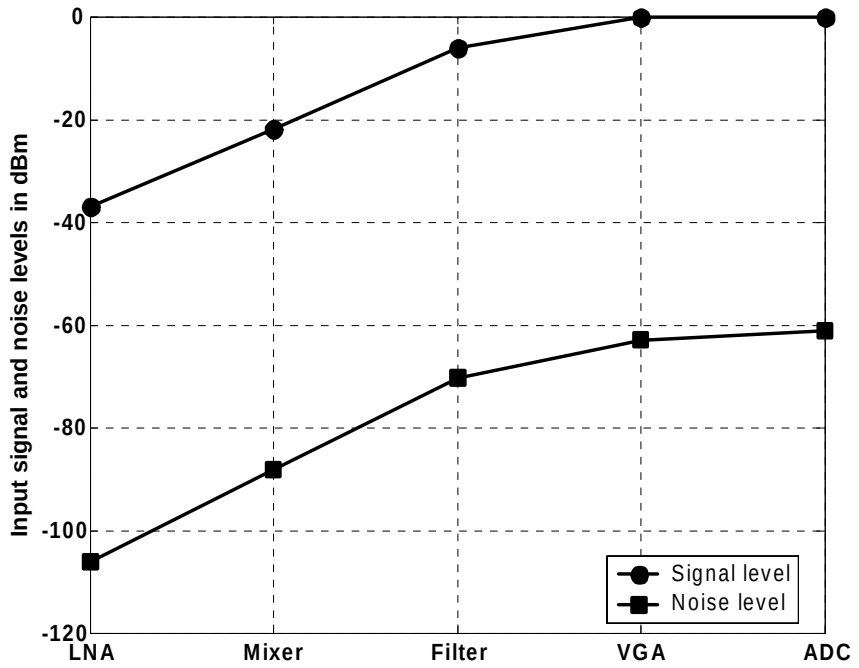


Fig. 5.52 Signal test at gain switching point in Wi-Fi mode (LNA gain is activated)

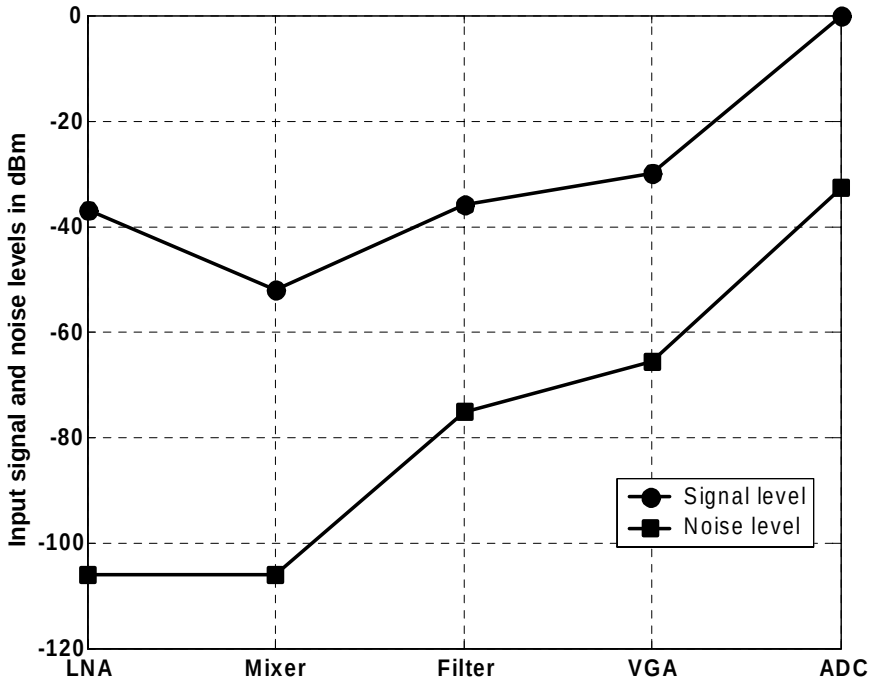


Fig. 5.53 Signal test at gain switching point in Wi-Fi mode (LNA is bypassed)

CHAPTER VI

CHAMELEON: CIRCUIT IMPLEMENTATIONS

In this chapter, the implementation of every building block, from LNA to ADC, in the dual mode Bluetooth/802.11b receiver will be presented. Each block was designed by one or two PhD students. More design details will be given for the variable gain amplifier (VGA) as it was my design assignment.

6.1. LNA and Mixer

The LNA circuit is shown in Fig. 6.1. It employs an inductively degenerated differential pair structure. LNA input matching is established on chip using inductive source degeneration technique [51]. In the presence of large signals, the LNA bias current I_{tail} is cut off and the signal is passed through a -15dB attenuator formed by MOS transistors M_5 – M_9 in the triode region. A capacitor C_m is inserted into the attenuator to improve impedance matching for the low gain mode. The only required off chip component is a 1:1 balun for single-ended to differential conversion. In the high gain mode, at small signal levels, all NMOS transistors of the attenuator M5 – M9 are turned off by connecting their gates to ground. Thus, the normal operation of the LNA is not affected. NMOS RF driving stage is used because MOS has better linearity than bipolar for the same bias current. NPN cascode transistors are used rather than NMOS transistors to have a more stable voltage at the drain of each of the NMOS RF driving transistors M_1 and M_2.

The mixer shown in Fig. 6.2 is a fully differential Gilbert-cell based structure. The I and Q branches sharing the same RF drive stage, therefore eliminating the RF drive stage mismatch present in the conventional two separate I/Q mixers. The current commutating switches are NPN bipolar transistors which require less LO power than NMOS transistors switch pairs. RF driving stage uses NMOS transistor for high linearity.

All of the biasing voltages and currents of the LNA and mixer are derived from a PTAT current source to ensure temperature stability of the RF front-end.

Measurement results for the input impedance matching condition for the high and low gain modes are shown in Figs. 6.3 and 6.4, respectively. In both cases, the input reflection coefficient (S_{11}) is less than -11dB. This means that, from the incident input RF power to the LNA, only -11dB of that power is reflected back due to imperfect matching. Summary of LNA/Mixer measurement results are listed in table 6.1.

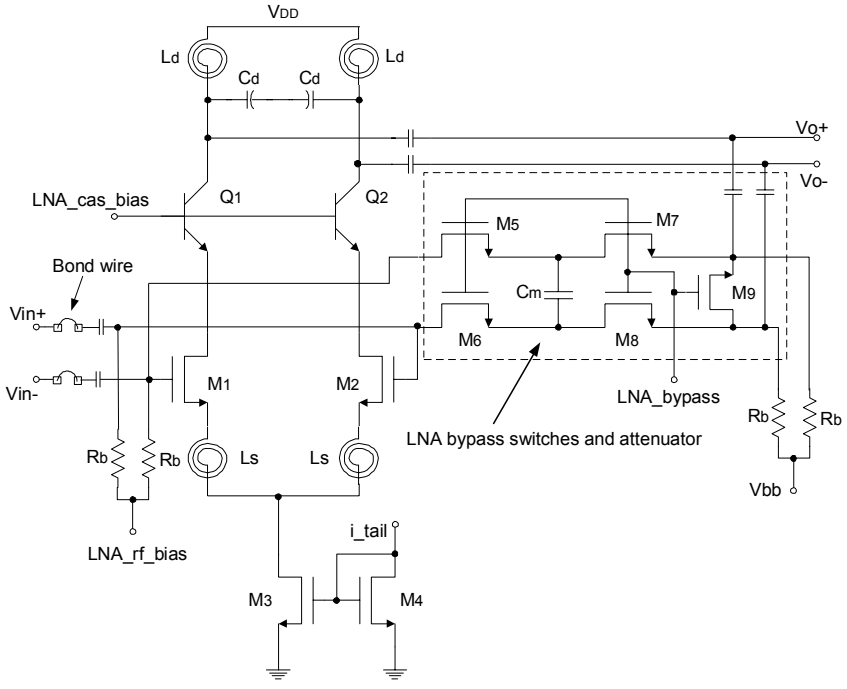


Fig. 6.1 LNA circuit

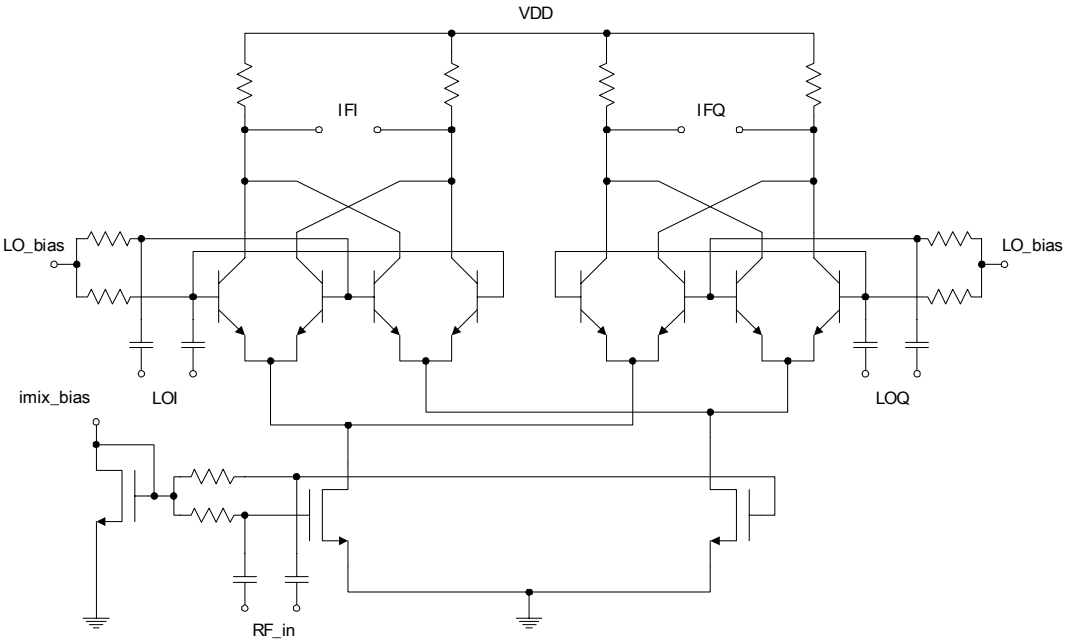


Fig. 6.2 Mixer circuit

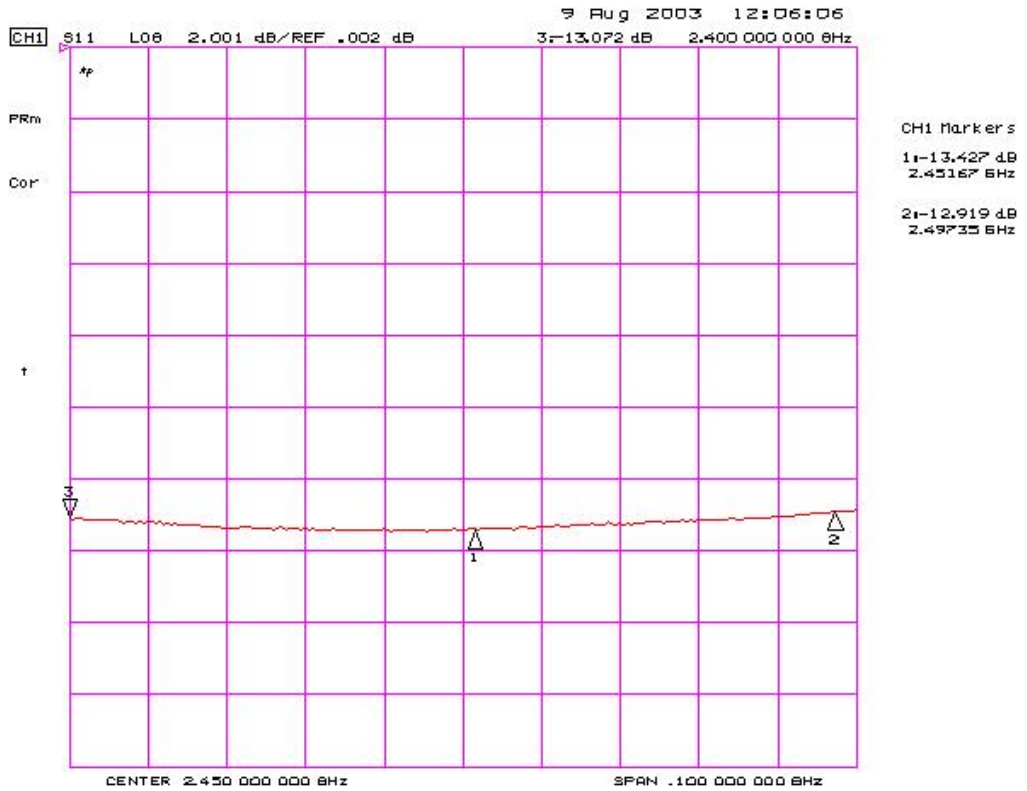


Fig. 6.3 Input matching for the high gain mode

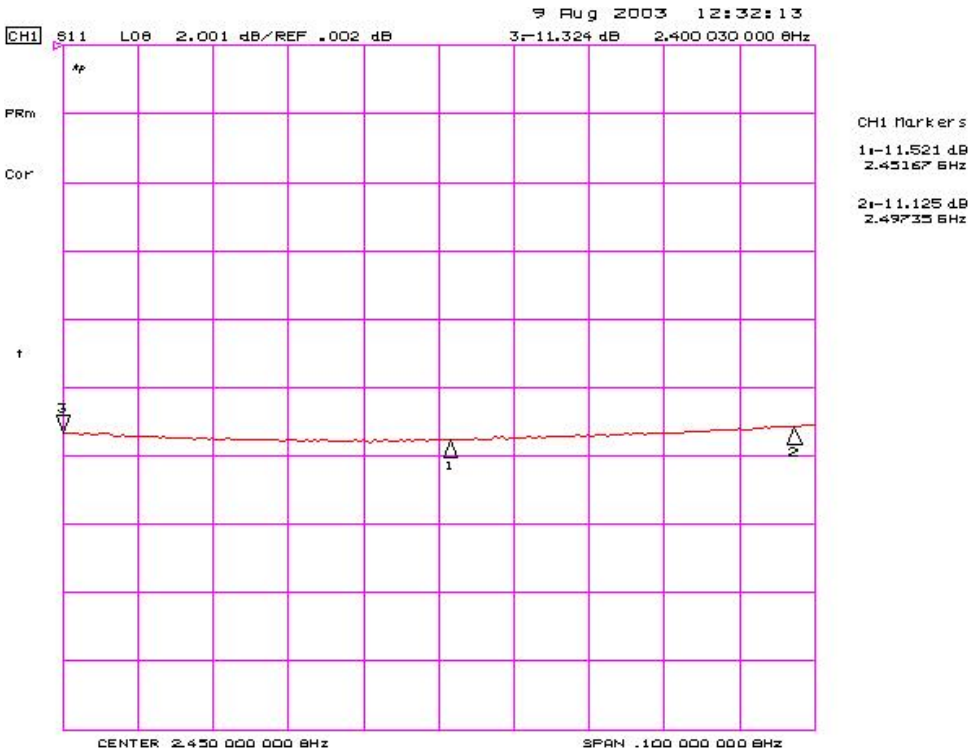


Fig. 6.4 Input matching for the low gain mode

6.2. VCO and PLL

The LO signal is generated by an integer-N type frequency synthesizer covering the specified band for both standards: from 2.4GHz to 2.5GHz in 1MHz steps. A typical dead-zone free phase-frequency detector (PFD), with 1ns output pulses in lock condition, is used. The PFD is followed by a charge pump (Fig. 6.5) with a cascode output [52]. The cascode transistors provide a larger output resistance that reduces the output voltage dependence of the output current. Switches M_{sp}–M_{sn} are sized to reduce the current mismatch and switching time of the charge pump. The charge pump current I_{cp} is 35μA.

A second order filter shown in Fig 6.6, uses a capacitance multiplier [53] to implement capacitor C_1 with a reduced area, minimum power consumption and negligible noise contribution. Leakage current can be a problem in the capacitance multiplier if the

attenuation of the loop filter at the reference frequency is not large enough. A large transistor length and small bias current are used to minimize the effect of the leakage current on the reference spurs.

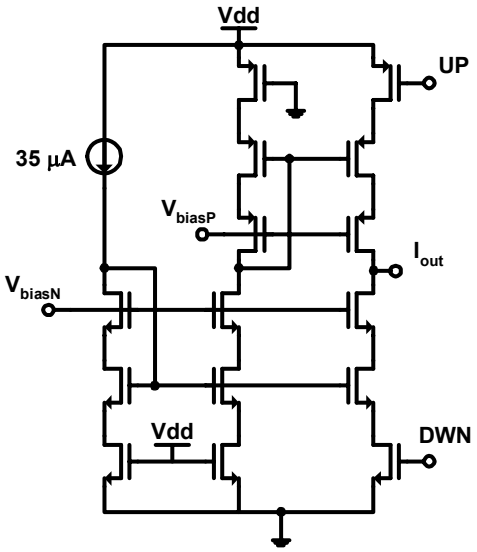


Fig. 6.5 Cascode charge pump schematic

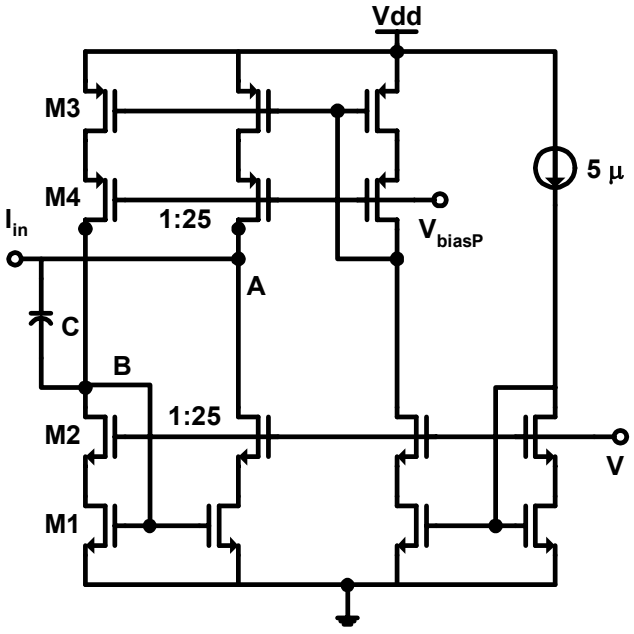


Fig. 6.6 Capacitance multiplier schematic

The VCO is operated at twice the LO frequency and the PLL is locked to a 2MHz external reference crystal oscillator. A divide-by-two flip-flop generates a pair of quadrature LO signals with a 1MHz tuning step. Even though more power is required to operate the VCO and prescaler at twice the desired LO frequency, two important features are improved under this strategy. First, the resultant phase and amplitude mismatch in the quadrature output of the divide-by-two is very low: measured as 1° in phase and 2% in amplitude. Second, the need for power hungry drivers for passive phase shifters is avoided. Furthermore, to run the VCO at a different frequency from the power amplifier of the transmitter is mandatory to avoid pull-in problems.

The VCO is implemented with a LC-tuned negative-g_m oscillator as shown in Fig. 6.7. Selected BiCMOS technology provides some unique options for designing the passive tank elements. Special low-resistance, top-metal layers is utilized for the on-chip inductor. Simulated quality factor of the 1.5nH inductor is 13. Intrinsic base-collector diode of bipolar device is used as a varactor, which provides ±17% capacitor variation range. The varactor can provide the VCO with 760MHz of tuning range: wide enough to overcome process and temperature variations. The inductor has a grid of deep trench underneath that helps to isolate it from substrate-coupled interference. Several measures have been taken to meet the phase noise requirement. Base nodes of bipolar transistor drivers are AC-coupled with oscillating nodes and biased by an extra DC biasing circuit to keep the transistors in the active region. Although the biasing circuit increases the effective base resistance, improved linearity helps to reduce the overall phase noise. Bypass capacitor on the common emitter node reduces noise the contribution of the

current bias transistors. The measured phase noise is -120dBc/Hz at a 3MHz offset and is plotted in Fig. 6.8.

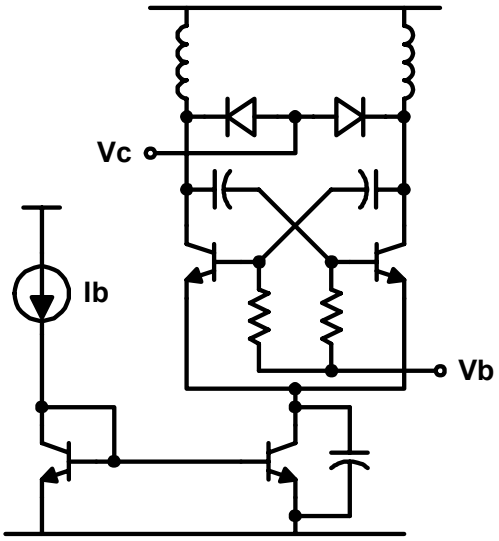


Fig. 6.7 VCO schematic

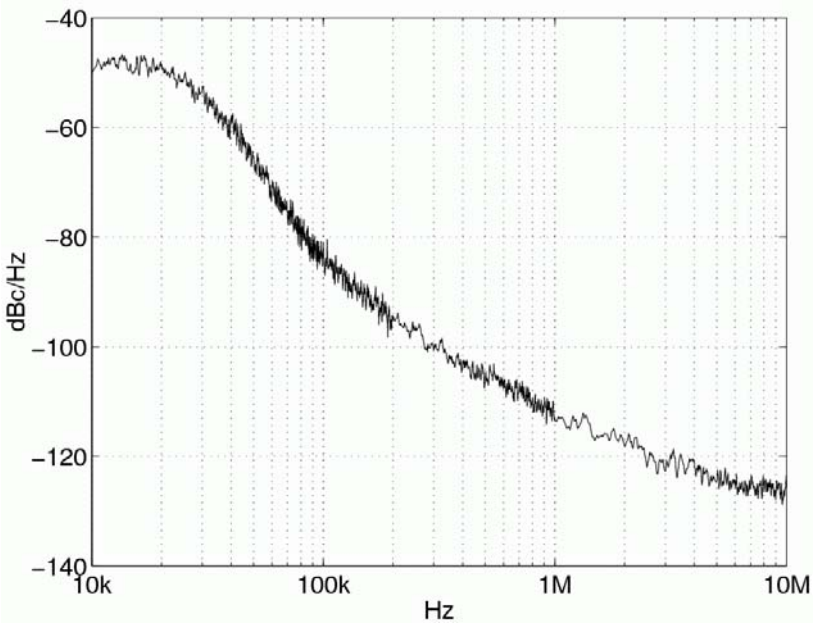
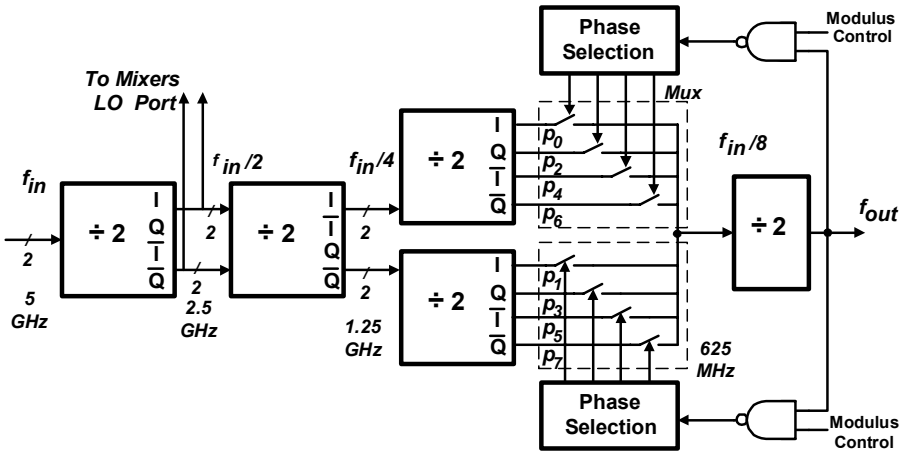
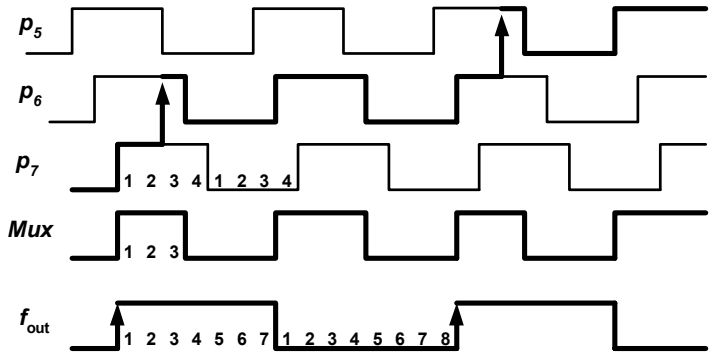


Fig. 6.8 Phase noise of the LO

A 15/16 dual-modulus phase switching prescaler follows the VCO [54]. The prescaler is comprised of three stages of cascaded asynchronous dividers, a 8-to-1 multiplexer, phase selection circuitry and a final divide-by-two stage, as shown in Fig. 6.9a. The use of asynchronous frequency dividers reduces the power consumption of the prescaler compared to conventional dividers operating at the same frequency, The output of the third stage of dividers generate eight phases separated by a 45° each at a frequency corresponding to 1/8 of the VCO frequency. When the modulus is set to 16 (Modulus Control = '0'), no switching occurs and a single phase goes through the phase selection block (Fig. 6.9b). When the modulus control signal is changed to 15 (Modulus Control = '1'), the phase selection block changes the output to the signal whose phase lags 45° the current signal in every output cycle (i.e. p7 $\rightarrow$ p6). The phase switching scheme is presented in Fig. 4b. Initially phase p_7 is selected at the output of the multiplexer, when a rising edge occurs in the output of the prescaler, the phase selector block changes the output of the multiplexer to p_6, which lags p_7 by 45°. By changing the output of the multiplexer from one phase to the one lagging it by 45°, the pulse width at the output of the multiplexer is reduced, corresponding to skip the count one input cycle and increasing the instantaneous division ratio.



a) Phase switching prescaler



b) Operation principle

Fig. 6.9 Phase switching prescaler

6.3. Channel Select Filter

The baseband 5th order LPF filter is implemented as an OTA-C structure with two cascaded biquadratic sections and a single passive pole, as shown in Fig. 6.10. A biquadratic section is shown in Fig. 6.11.

The proposed OTA architecture for the dual-mode filter is shown in Fig. 6.12. The OTA used is a source degenerated transconductance, to enhance the limited linear input range of the bipolar differential pair, with current scaling. The filter needs to provide a

programmable cut-off frequency ratio of 10:1. To avoid components values spread, both resistors and capacitors are scaled by factors of 5 and 2, respectively.

Since the filter linearity requirement is almost the same in both modes, the bias current is inversely scaled with the same factor as the resistance to keep the product $I_{bias}\times R$ constant. This helps to optimize the power consumption in both modes. The active loads M2 are also scaled accordingly by increasing the width of the load transistors so as to keep the source-gate voltage constant in the two modes.

The first order passive pole used at the input stage of the filter helps improving the out-of-band linearity. This improvement is due to the attenuation of the undesired interferers. This is done at no cost in power. The cost is rather in area and noise. From simulation results, adding the first order pole improves the out-of-band IIP3 by about 10dB and deteriorates the NF by about 1.5dB.

Lack of tracking between the single pole resistors and the transconductances of the Gm cells, would result in pass-band ripple [55]. The transconductance of the source degenerated OTA, shown in Fig. 6.12, is expressed as:

$$G_m = \frac{1}{R}\times\frac{g_{m1}R}{1+g_{m1}R} = \frac{1}{R}\times\frac{n}{1+n} \qquad \text{where:} \quad n = g_{m1}R \tag{6.1}$$

In order to minimize mismatches between the filter active poles and the first passive pole, variations in the factor n/(n+1) in equation (6.1) must be minimized. Relative variations in G_m can be expressed in terms of variations in n as:

$$\frac{\Delta G_m}{G_m} = \frac{1}{1+n}\left(\frac{\Delta n}{n}\right) \tag{6.2}$$

Therefore, in order to reduce the effect of process variations on the filter transfer function, n should be increased. For example, if $n = g_{m1}R = 10$, 50% process variations

in n corresponds to about 4.5% variations in G_m. It has been verified from simulations that this ripple is less than 0.2dB which is acceptable for the current application.

Since the out-of-band blockers will be attenuated by the filter, harmonics generated by the out-of-band blockers are dominated by the filter's first stage. Hence, to improve the overall filter linearity, the first stage is designed to have better linearity by using more current I_{bias} in the first stage. Increasing the current of the first stage by 50% (the total current by 10%) results in a 3dB improvement in IIP_3 of the overall filter.

The dimensions of the switches in series with the degeneration resistors are chosen to make the switch resistance about 1% of the resistor such that the linearity performance of the OTA, and consequently the filter, is almost not affected.

The filter dissipates 2.7mA and 0.9mA for Wi-Fi and Bluetooth modes, respectively. It has 6dB gain and occupies $0.9mm^2$ of silicon area. The measured frequency response of the filter in both modes and for different tuning control bits is shown in Fig. 6.13. Measured filter characteristics are summarized in table 6.1.

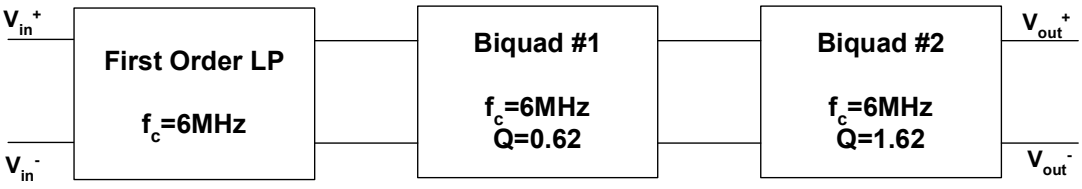


Fig. 6.10 Butterworth filter block diagram

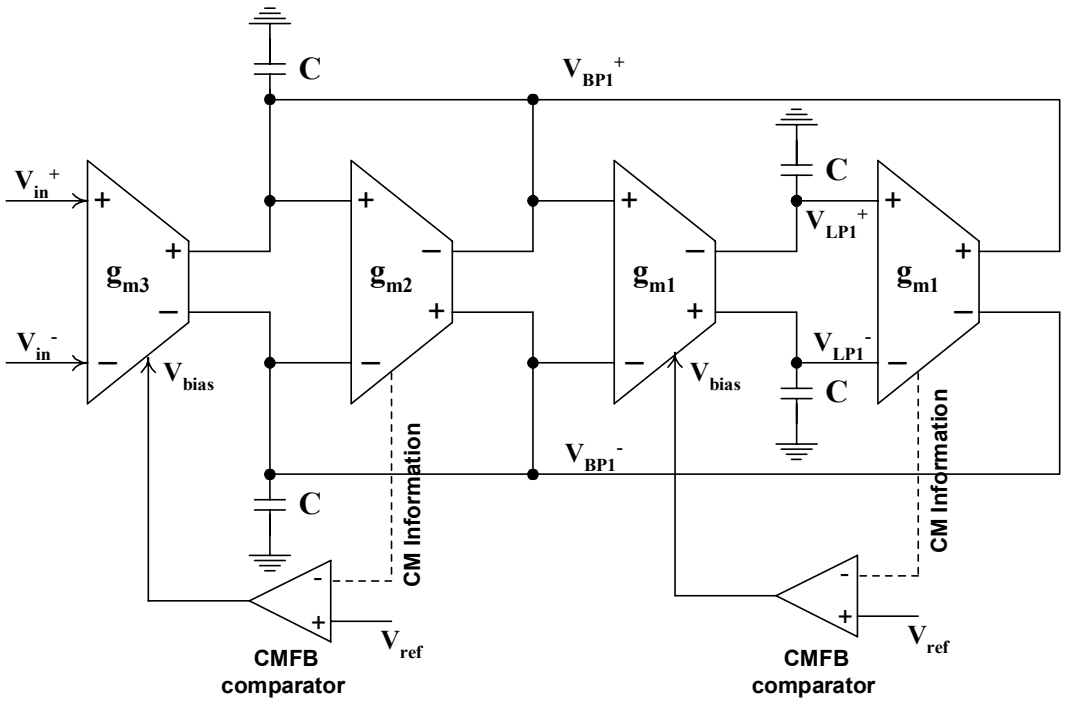


Fig. 6.11 Biquadratic section

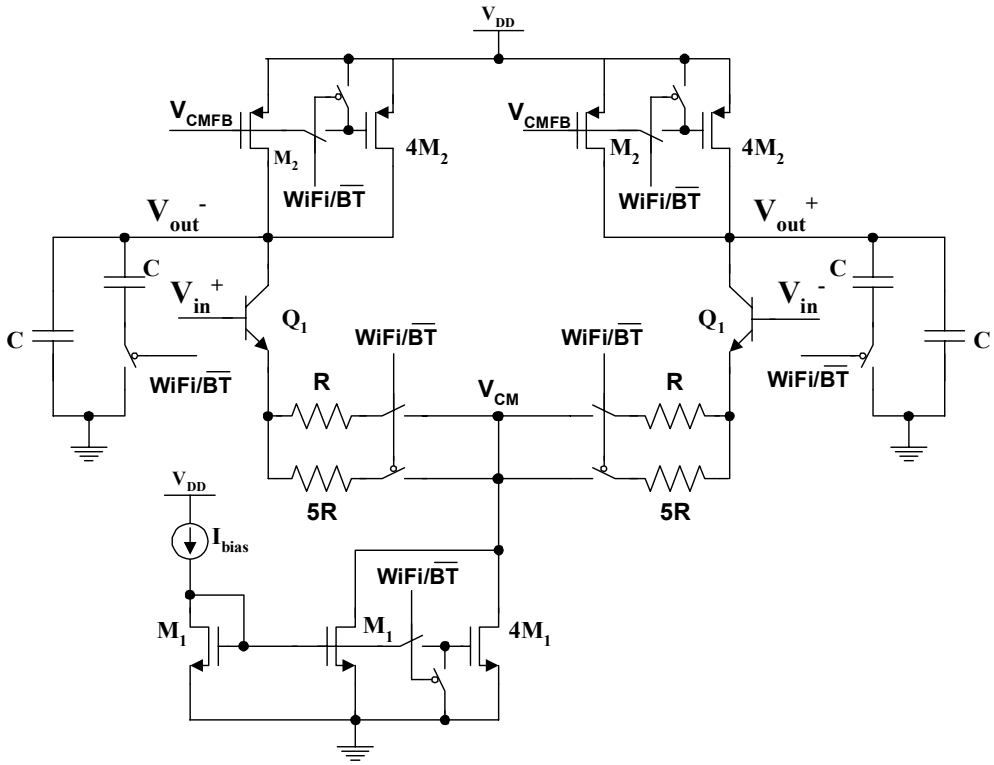


Fig. 6.12 Implementation of the dual-mode OTA

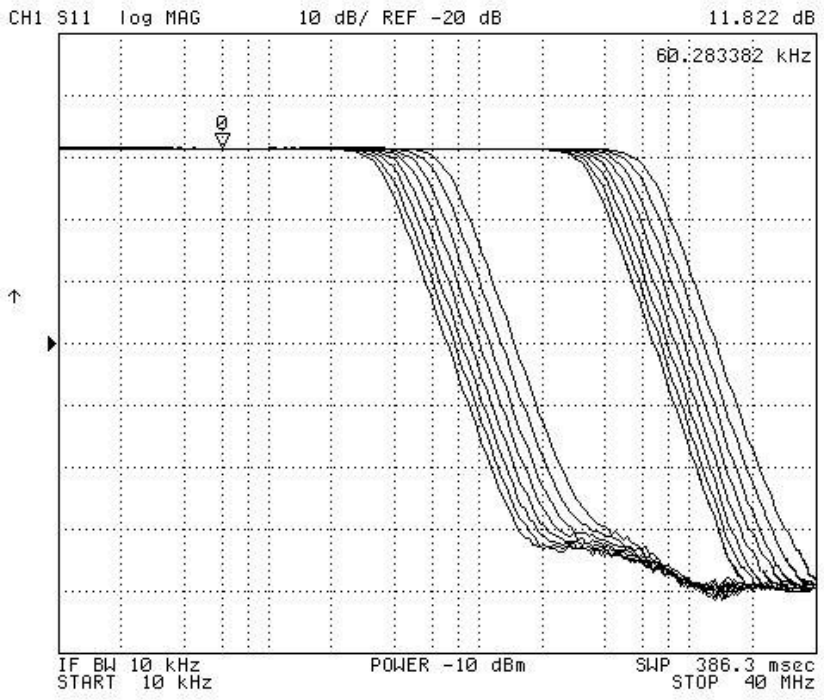


Fig. 6.13 Dual-mode operation and programmability of the LPF

Table 6.1 Summary of performance

	Bluetooth mode	**802.11b mode**
Filter type	Butterworth	Butterworth
Filter order	5	5
Cut-off frequency	600kHz	6MHz
IIP_3	40dBm [1]	40dBm [2]
Integrated input referred noise	$90\mu V_{rms}$	$100\mu V_{rms}$
Power supply	2.5V	2.5V
Current consumption [4]	0.9mA [3]	2.7mA [3]
Area [4]	$0.9mm^2$	$0.9mm^2$

[1] Two tones applied at 1.83MHz and 3.16MHz
[2] Two tones applied at 18.3MHz and 31.6MHz
[3] Including CMFB and biasing circuitry
[4] I and Q branches

6.4. Variable Gain Amplifier

6.4.1. DC Offset Problem, Revisited

If the receiver has no prior information regarding the incoming signal level, it has to figure it out by itself when it receives the first bits of the incoming stream and then adapts itself as fast as possible to receive and decode the information. The overall receiver gain is adjusted, depending on the signal level, to keep almost constant signal level at the ADC input through the VGA [56-60]. The most important factor that determines the response time of the receiver is the settling time of its VGA, which is inversely proportional to the VGA bandwidth. A number of wideband VGA designs can be found in the literature [57, 60]. The problem becomes even more difficult for direct-conversion receivers where the dc offset due to mismatches is a serious issue. In general, the output dc offset of a VGA depends on its gain, and therefore, has to be corrected each time the VGA gain is changed. Another requirement in the VGA is to have constant bandwidth (and therefore, constant settling time) versus gain. Table 6.2 lists a summary of required specifications for the VGA.

Table 6.2 Summary of required VGA specifications

Specification	Value	
	Wi-Fi	Bluetooth
Gain	0-62dB	0/24dB
Gain step	2dB	24dB
Bandwidth	>10MHz	>3MHz
Input referred noise	$<28.2nV/\sqrt{Hz}$	$<50.3nV/\sqrt{Hz}$
1dB compression point	$>2V_{pp}$	$>2V_{pp}$

The BiCMOS process is used to save power consumption in the multi-standard receiver. However, the VGA design has been done using only CMOS transistors which means that it can also be implemented in a CMOS process.

Since the maximum gain of the VGA is 62dB in Wi-Fi mode, a 3-stage VGA is used with about 20dB gain/stage. However, only the 3rd stage is used in Bluetooth mode to provide 0/24dB gain. The VGA is used in a direct conversion receiver, and therefore, dc offset generated by mismatches in the baseband blocks (including the VGA itself) may saturate the stages of the VGA unless this offset is cancelled before it gets amplified by the VGA stages. The offset can be cancelled between VGA stages by injecting a correction signal from a digital-to-analog converter (DAC) as illustrated in Fig. 6.14 or by using simple RC high-pass filters with low cutoff frequency as in Fig. 6.15. The cutoff frequency must be chosen carefully: too low a cutoff might not remove enough 1/f noise, whereas too high a cutoff may remove precious signal spectrum. According to BER baseband simulations, 5kHz cutoff results in only 0.2dB degradation in the required SNR_{min} in Wi-Fi mode. In Bluetooth mode, 1kHz cutoff results in 0.1dB SNR_{min} degradation. The HPF solution will only work if the dc offset remains constant while the signal is being received. However, since the gain of the VGA is adjusted during the preamble period, the output offset of each VGA stage may change if its gain is altered. This is because the capacitor in the HPF takes time (which is inversely proportional to the HPF bandwidth) to charge/discharge to the new offset value.

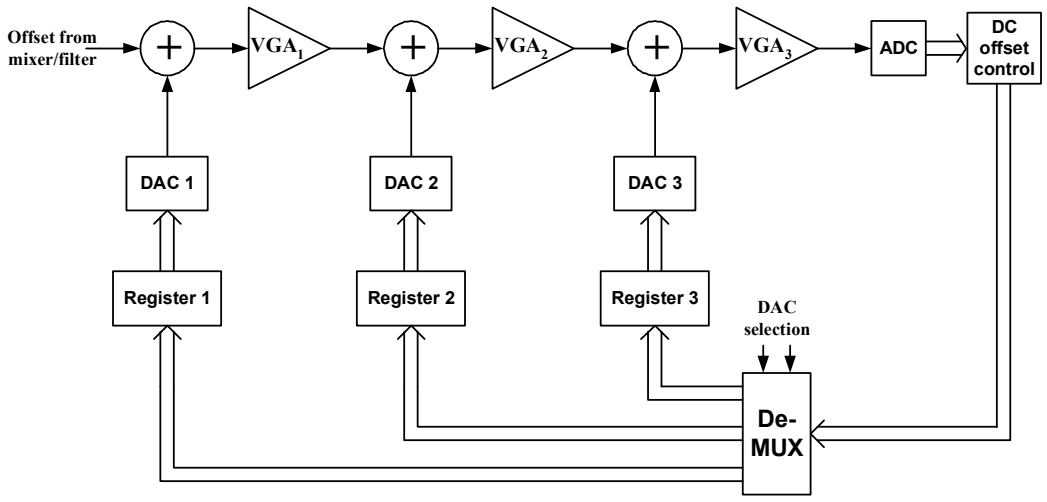


Fig. 6.14 DC offset correction using digital feedback

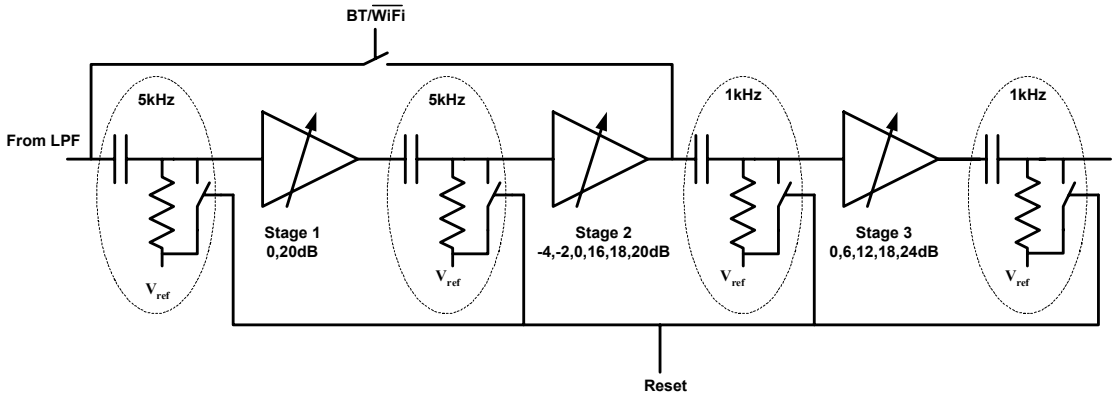


Fig. 6.15 DC offset correction using passive HPF

6.4.2. Proposed Solution for HPF Slow Response

The proposed solution to the slow response of the HPF to dc offset variations is to design the VGA stages such that its output referred offset is constant regardless of its gain setting. To illustrate how to design such a VGA, consider first the OpAmp-R based VGA shown Fig. 6.16(a). Single ended topology is shown for simplicity. Differential architecture is used in all VGA stages. The OpAmp input referred offset is represented by

the voltage source V_{OS} at the non-inverting terminal. The gain of the VGA stage is controlled by the digital bit d_1. The output of the VGA in Fig. 6.16(a) is given by:

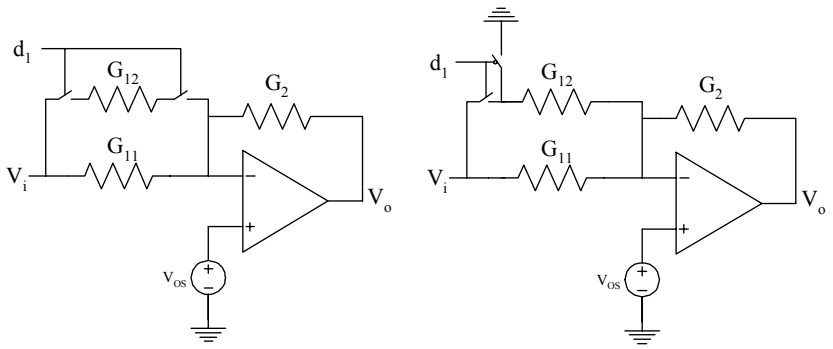
$$V_{out} = -\frac{G_{11} + d_1 G_{12}}{G_2} V_{in} + (1 + \frac{G_{11} + d_1 G_{12}}{G_2}) V_{OS} \tag{6.3}$$

Note that the second term in the above equation, which represents the output referred offset, depends on the digital control input d_1. Therefore, the output dc offset of this VGA circuit depends on its gain. To keep the output dc offset independent of gain, the VGA stage shown in Fig. 6.16(b) is proposed. The output voltage of the modified VGA stage is expressed as:

$$V_{out} = -\frac{G_{11} + d_1 G_{12}}{G_2} V_{in} + (1 + \frac{G_{11} + G_{12}}{G_2}) V_{OS} \tag{6.4}$$

In this case, the resistor G_{12} is always included in the feedback loop and therefore, the output referred offset is constant as indicated in equation (6.4). The gain is changed by switching only the input terminal of the resistor G_{12} to the input or to ac ground. Note that the output dc offset is constant only if the offset from the previous stage is completely rejected. The circuit in Fig. 6.16(b) has constant feedback factor regardless of its gain. This means that the bandwidth of the VGA will be independent of its gain. This is a very important property in a VGA design. The main drawback in this circuit is its finite input resistance which is also variable. Since each VGA stage is preceded by a passive HPF that uses large resistance and capacitance values to achieve the required low cut-off frequency, the output of this passive HPF has to be buffered before it's applied to the VGA input. The buffer is implemented using a source follower with reduced output resistance (to be discussed later) and therefore can generate its own offset that will be multiplied by the variable gain of the VGA and produce a variable offset at the VGA

output. To circumvent this problem, a separate buffer is used for each resistor in the bank of resistors at the VGA input and the gain is controlled by the switching the input of the buffer, not the output as shown in Fig. 6.17. Therefore the buffer offset is always included in the circuit and the VGA output offset is kept constant.



(a) (b)

Fig. 6.16 (a) Conventional OpAmp-R VGA (b) Proposed constant output offset VGA

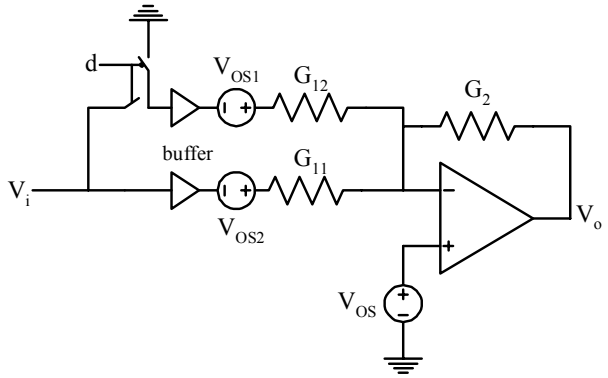


Fig. 6.17 Proposed constant output offset VGA circuit using OpAmp-R

6.4.3. OpAmp and Buffer Designs

To achieve a gain of about 20dB in a VGA stage, a 3-stage OpAmp architecture has been chosen, as shown in Fig. 6.18 The input and output common mode voltages are set to 1.5V. Simulations showed that the OpAmp dc gain is more than 70dB and the phase

margin is more than 60° at 20dB gain. The 3rd stage of the OpAmp is the buffer stage, which drives only the feedback resistor G_2 as shown in Fig. 6.17, and the resistors R_{CM} used to sense the common mode output of the OpAmp in the common mode feedback (CMFB) circuit (Fig. 6.19). The compensation capacitor C_C is used to stabilize the differential mode (DM) and common mode (CM) loops. Although the OpAmp is always used in a high DM gain configuration, the CM loop uses unity gain and its stability has to be enhanced by adding a degeneration resistor R_{CMFB} in the CMFB transconductor shown in Fig. 6.19. It's also worth to emphasize the advantage of using the configuration in Fig. 6.16(b) that the feedback factor is always constant and therefore, the OpAmp will have the same phase margin for all gain settings. This allows us to use fixed compensation capacitor while keeping the OpAmp bandwidth constant for all VGA gains. The buffers though, have to drive the input resistors of the VGA stage which are smaller or equal to the feedback resistor. Since inter-stage HPF's are used, the input CM voltage of the buffers doesn't have to be the same as the output CM voltage, and therefore simpler buffers with dc level shift can be used. To improve linearity and gain accuracy, a buffer with reduced output resistance, as shown in Fig. 6.20, is used. The low frequency output resistance of the buffer is approximately given by:

$$R_{out} = \frac{g_{d1} + g_{d3}}{g_{m2} g_{m3}} \tag{6.5}$$

Where g_{mi} and g_{di} are the small signal transconductance and output conductance of the transistor M_i.

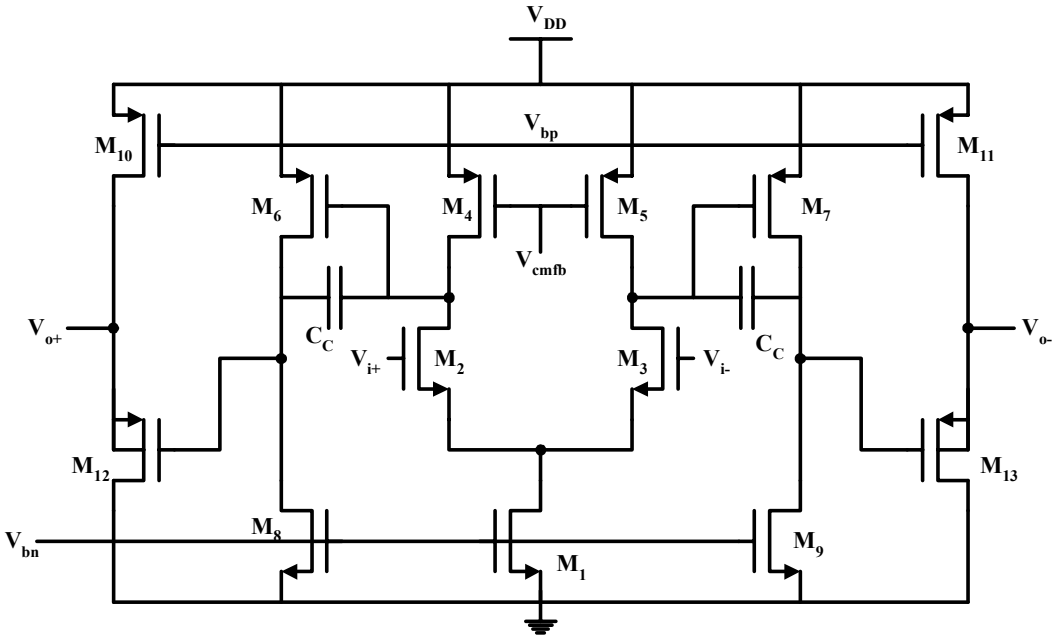


Fig. 6.18 3-stage OpAmp schematic

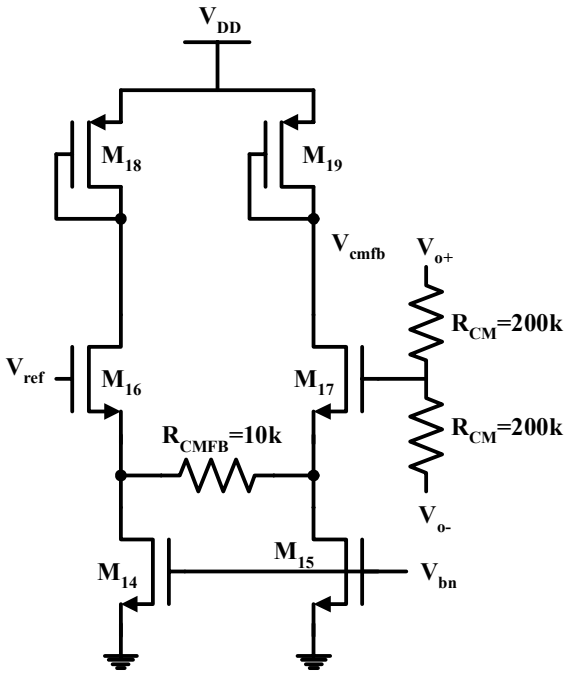


Fig. 6.19 CMFB circuit

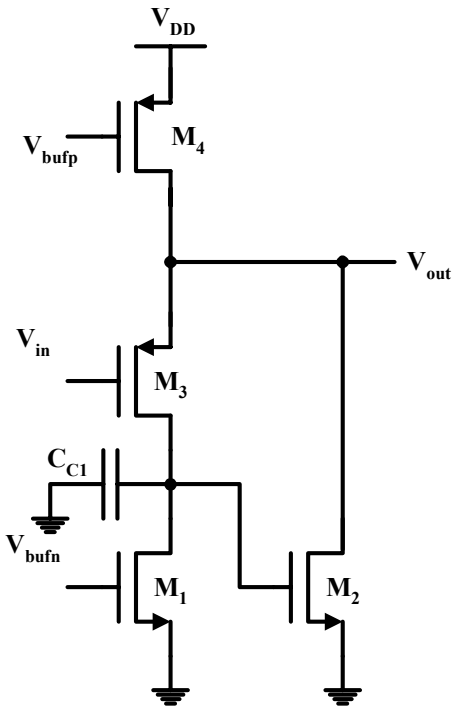


Fig. 6.20 The buffer circuit

6.4.4. VGA Gain Lineup

The VGA gain has been distributed among the 3 stages as shown in Fig. 6.15. The gain si distributed such that all the stages have almost the same maximum gain (about 20dB gain/stage). In the first stage, the gain can be either 0 or 20dB. Fine gain control is achieved through the second and third stages. The third stage has 5 possible gain settings from 0 to 24dB with 6dB gain steps. The second stage has 6 possible gain settings which are -4, -2, 0, 16, 18, 20dB. This gain control distribution allows the overall VGA gain to be controlled from -4 to 64dB with 2dB steps. The gain control is distributed such that the number of gain steps/stage is reduced to avoid the design complexity in each stage. 6dB steps are used in the last stage to allow using R2R ladder network for gain control as will be discussed later. The following sections describe the design of the VGA stages.

6.4.4.1. First stage

The first stage of the VGA has the most stringent requirements in terms of noise and linearity. This is because the noise of this stage is multiplied by the gain of the other two stages. On the other hand, the minimum gain for this stage is 0dB, which means that the signal can be at full scale voltage ($1V_p$). The first stage VGA circuit is similar to the circuit shown in Fig. 6.17 with $G_2 = G_{11} = G_{12}/9 = 1/50k\Omega$. Again, single-ended topology is shown for simplicity. The gain of the first stage is 0dB when d_1=0 and 20dB when d_1=1.

6.4.4.2. Second stage

The second stage of the VGA has more relaxed noise requirements than its first stage. Therefore, in order to reduce the power consumption in the OpAmp, the input and feedback resistors are scaled up to reduce the loading on the OpAmp. To have about the same circuit bandwidth, all the transistors sizes as well as dc currents are scaled down by the same factor. Therefore, the power consumption of the second stage is lower than that of the first stage by the same factor. Since the resistors values in the first stage are already high (50kΩ), the resistor values in the second stage cannot be increased much higher because of area limitation. Fig. 6.21 shows the second stage VGA circuit. The gain of the second stage is controlled with the digital inputs d_2 to d_5. Gain values and the corresponding digital inputs are listed in the table 6.3.

The values of R_{i1}, R_{i2}, and R_{i3} (where i=1,2) are chosen such that:

$$20\log\left(\frac{R_{i1}}{R_{i1}+R_{i2}+R_{i3}}\right) = -4dB \qquad (6.6)$$

$$20\log\left(\frac{R_{i1}+R_{i2}}{R_{i1}+R_{i2}+R_{i3}}\right)=-2dB \tag{6.7}$$

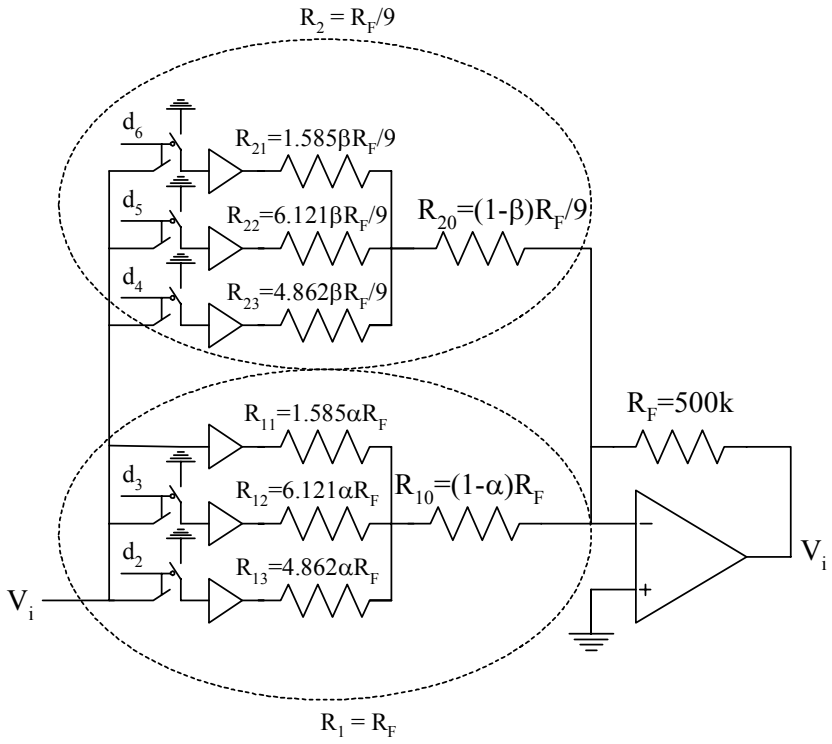


Fig. 6.21 Second stage VGA circuit

Table 6.3 Gain values of the second stage and the corresponding digital inputs

G_2 in dB	d_2	d_3	d_4	d_5	d_6
-4	0	0	0	0	0
-2	0	1	0	0	0
0	1	1	0	0	0
16	0	0	0	0	1
18	0	1	0	1	1
20	1	1	1	1	1

The parameters α and β in Fig. 6.21 are used to scale the input resistance seen by the buffers. Note that the equivalent resistance of the resistors R_{i1} (i=0, … , 3) is equal to R_F and is independent of the value of α. Similarly, the equivalent resistance of the resistors R_{i2} (i=0, … , 3) is equal to $R_F/9$ and is independent of β. The lower the values of α and β, the lower the total resistance area, and the higher the required driving capability in the buffers, and therefore the higher the current drain. This is an example of power-area trade-off in this design. Note that in order to have positive resistance values, α and β are positive numbers less than 1.

6.4.4.3. Third stage

The third stage of the VGA is designed to have 6dB gain steps from 0dB to 24dB. Therefore, a resistive ladder network can be used at the input to realize the required gains. Fig. 6.22 shows the third stage VGA circuit. Gain values and the corresponding digital inputs are listed in the table 6.4.

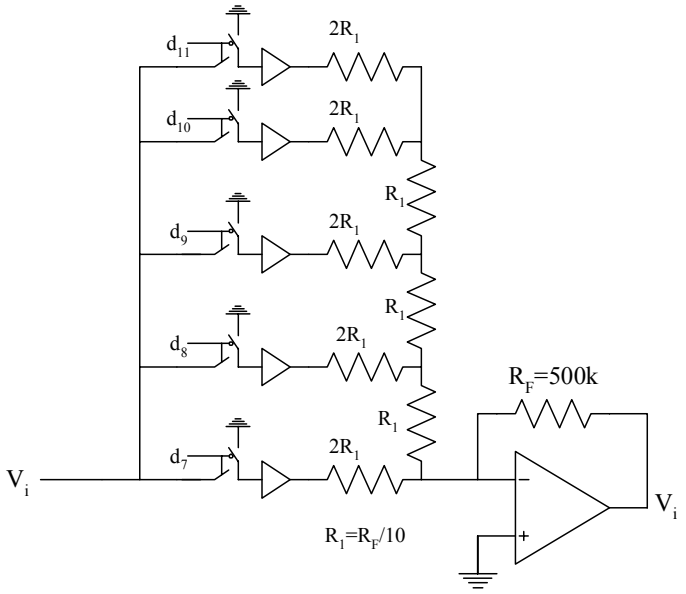


Fig. 6.22 Third stage VGA circuit

Table 6.4 Gain values of the third stage and the corresponding digital inputs

G_3 in dB	d_7	d_8	d_9	d_{10}	d_{11}
0	0	0	0	1	0
6	0	0	1	0	0
12	0	1	0	0	0
18	1	0	0	0	0
24	1	1	1	1	1

6.4.5. Testing Results

Since there are 35 VGA gain settings in the -4 → 64dB gain range, 6 control bits are needed to control the gain. A decoder should be built to decode the gain control bits into the d_i (i=1, … , 11) bits required for the switches in the VGA stages. One bit control is saved by using only the gain range 0 →62dB which is required for the Wi-Fi standard. Therefore, only 5 control bits a_m (m=0, … , 4) are used in the design. Table 6.5 lists the gain distribution over the three VGA stages for the 32 possible VGA gain settings. The VGA has been integrated with the other blocks in the multi-standard BT/Wi-Fi receiver. A single VGA branch consumes 1.05mA and occupies 0.5mm×0.7mm of silicon area. Testing setup of the VGA is illustrated in Fig. 6.23. The measured input referred noise density is $26nV/\sqrt{Hz}$ at maximum gain. Fig. 6.24 shows the time domain response of the VGA when an input signal of 40mV amplitude is applied. The gain of the VGA is switched from 4 → 20 → 28 → 12 → 4dB using the digital control bits. Fig. 6.25 shows the VGA settling time by zooming in the 20 → 28dB transition. The figure shows that the VGA settling time is less than 200ns. Fig. 6.26 shows the frequency response of the VGA for gain settings from 6 → 46dB with 10dB step. The VGA bandwidth is about 21MHz

and it is almost constant throughout the entire gain range. Measured input referred 1dB compression point at 0dB gain is -0.5dBm. Summary of VGA experimental results is listed in table 6.6.

Table 6.5 Gain distribution over the three VGA stages

Total Gain	1st stage gain (dB)	2nd stage gain (dB)	3rd stage gain (dB)	a_4	a_3	a_2	a_1	a_0
0	0	0	0	0	0	0	0	0
2	0	-4	6	0	0	0	0	1
4	0	-2	6	0	0	0	1	0
6	0	0	6	0	0	0	1	1
8	0	-4	12	0	0	1	0	0
10	0	-2	12	0	0	1	0	1
12	0	0	12	0	0	1	1	0
14	0	-4	18	0	0	1	1	1
16	0	-2	18	0	1	0	0	0
18	0	0	18	0	1	0	0	1
20	0	-4	24	0	1	0	1	0
22	0	-2	24	0	1	0	1	1
24	0	0	24	0	1	1	0	0
26	0	20	6	0	1	1	0	1
28	0	16	12	0	1	1	1	0
30	0	18	12	0	1	1	1	1
32	0	20	12	1	0	0	0	0
34	0	16	18	1	0	0	0	1
36	0	18	18	1	0	0	1	0
38	0	20	18	1	0	0	1	1
40	0	16	24	1	0	1	0	0
42	0	18	24	1	0	1	0	1
44	0	20	24	1	0	1	1	0
46	20	20	6	1	0	1	1	1
48	20	16	12	1	1	0	0	0
50	20	18	12	1	1	0	0	1
52	20	20	12	1	1	0	1	0
54	20	16	18	1	1	0	1	1
56	20	18	18	1	1	1	0	0
58	20	20	18	1	1	1	0	1
60	20	16	24	1	1	1	1	0
62	20	18	24	1	1	1	1	1

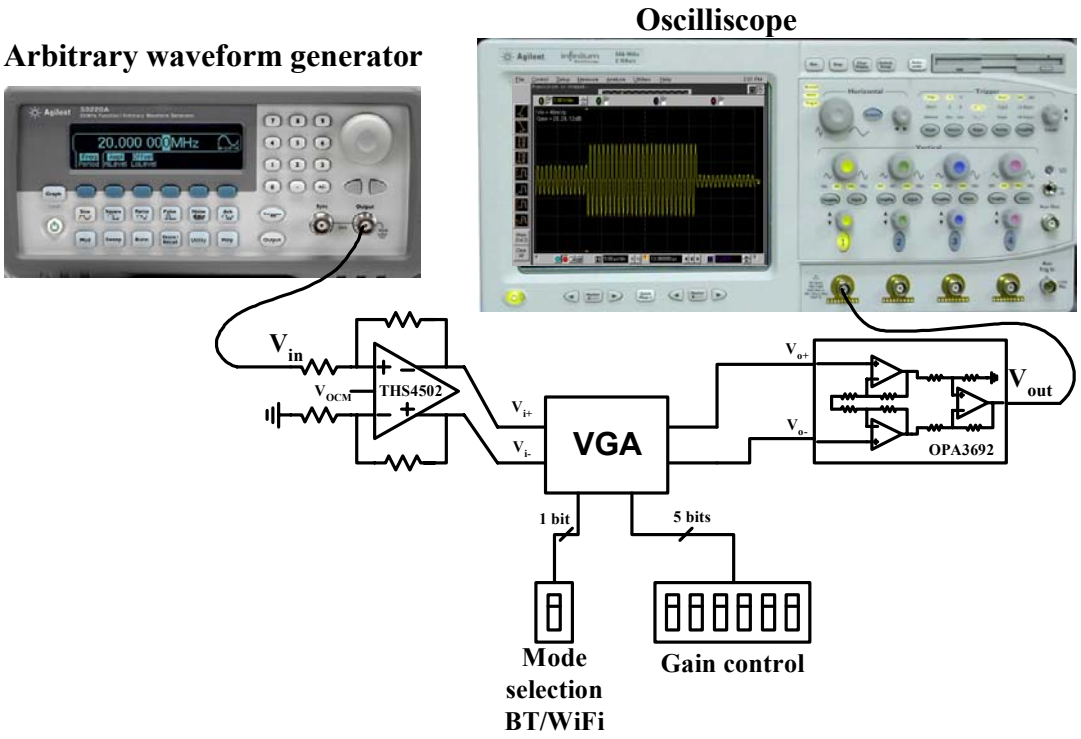


Fig. 6.23 VGA test setup

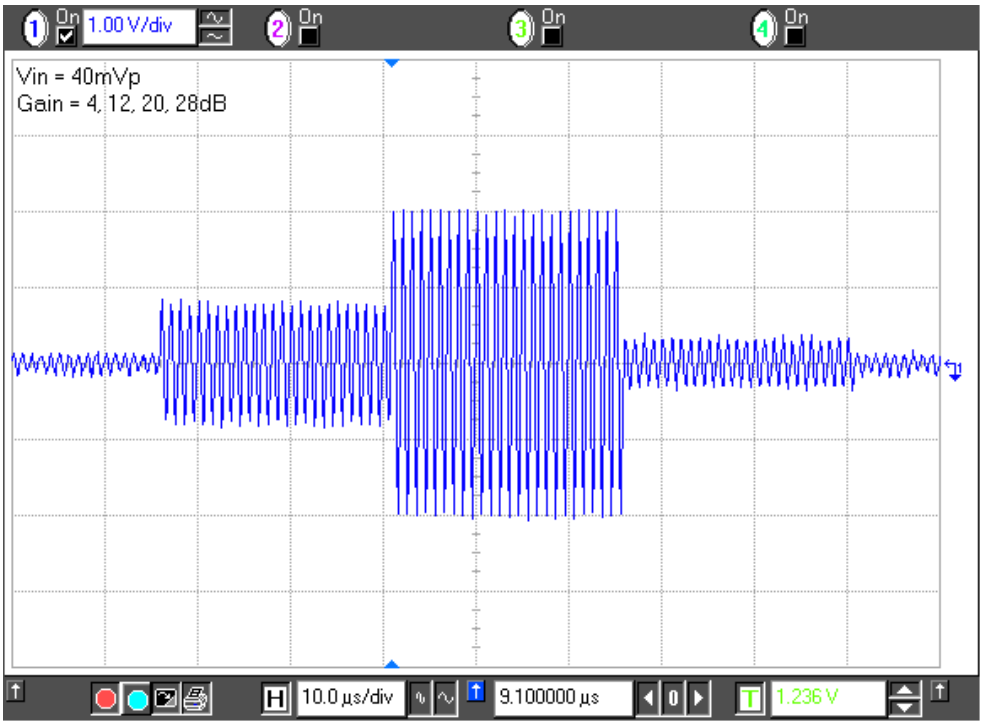


Fig. 6.24 Time domain response of the VGA

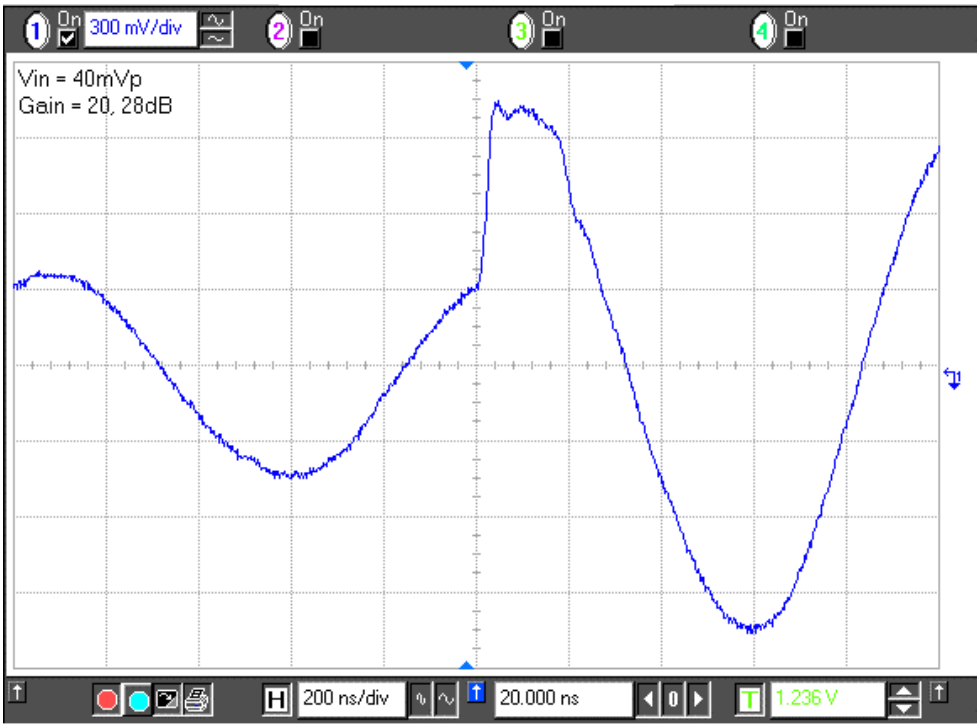


Fig. 6.25 Settling time of the VGA

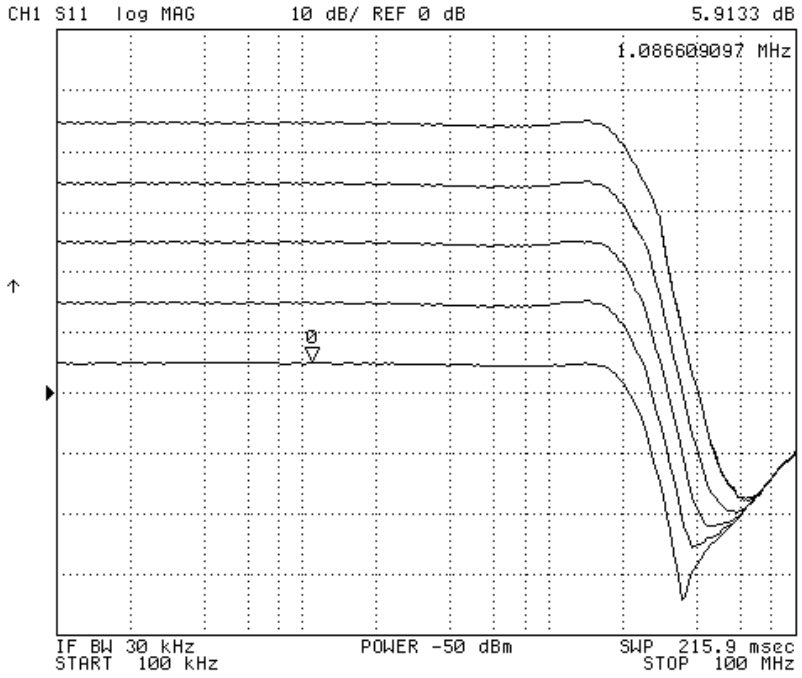


Fig. 6.26 Frequency response of the VGA at different gain settings

Table 6.6 Summary of VGA experimental results

Parameter	Value
Gain	0-62dB
Gain step	2dB
Bandwidth	21MHz
Current drain	1.05mA
Input 1dB compression (0dB gain)	-0.5dBm
Settling time	200ns
Input referred noise (62dB gain)	$26nV/\sqrt{Hz}$
Area	$0.35mm^2$
Power supply	2.5V

6.5. Analog to Digital Converter

As mentioned in the system level considerations, the ADC must provide a 62dB dynamic range at 11MS/s, and a 48dB dynamic range at 44MS/s for the Bluetooth and 802.11b base-band signals, respectively. So far, the only reported ADC architecture for multi-standard radios has been ΣΔ with adjustable OSR [61]. Due to the relatively wide signal bandwidth of the Wi-Fi signal at 11Mbps, the use of a ΣΔ ADC in this receiver would result in unacceptably high power consumption. On the other hand, a time-interleaved Nyquist-rate ADC can provide a consistent dynamic range over a wide frequency band while the overall ADC sampling rate can be configured by the number of the interleaved ADC branches in operation. In addition, the dynamic range can be controlled by the number of active stages if pipeline architecture is used. For this reasons, time-interleaved pipeline architecture has been chosen to implement a low-power configurable ADC for the Chameleon receiver.

The ADC architecture is shown in Fig. 6.27. It consists of a single front end sample-and-hold and two time interleaved 11 bit pipeline ADCs with alternate 4 and 3 bit stages. In the Wi-Fi mode, both pipeline branches operate at 22MHz to provide an overall

sampling frequency of 44MHz. The last stage in each pipeline branch can be disabled in this mode to reduce power consumption while still providing the required dynamic range. For the BT receiving mode, only one pipeline branch is activated and the sampling rate is scaled down to 11MS/s. A multi-bit pipeline architecture has been chosen instead of the traditional 1.5bit/stage design to reduce the number of MDACs, the implementation cost of the lower bits stages and hence the power consumption. As an additional advantage, the use of multi-bit stages significantly reduces the ADC delay, which is convenient for the real-time operation of the receiver.

Several design measures were taken at the circuit-level design to optimize the power consumption. OpAmp sharing technique, instead of two individual OpAmps, is applied in the S&H. CMOS and Bipolar transistors are combined in the OpAmps of the S&H and MDAC to achieve the requirements with minimum static bias current. The 3-bit sub-ADC employs a flash architecture while the 4-bit ADC adopts 2X flash interpolation structure to reduce the number of the preamps. The sampling circuit at the input of the preamp is designed to avoid loading the MDAC of the previous stage, thus significantly reducing the current consumed in capacitance charging. Bipolar transistors are employed in the comparator to achieve the desired speed using minimum static current.

The digital correction in the pipeline and the multi-bit structure relax the sub-ADCs specifications and gain error requirement. However, calibration is still needed to maintain the 11-bit linearity in the first stage MDAC. Fig. 6.28 depicts the online digital calibration scheme applied. The principle of the digital calibration has been introduced before [62], but the calibration has to be performed offline, which is undesirable in a receiver. In this design, an identical pipeline ADC branch is added to the existing parallel

ADCs. The first stage MDACs are calibrated one by one in rotation. The first stage of the extra branch substitutes and functions as the stage in calibration. Thus, the calibration is transparent to the circuit outside. All the first stage MDACs are calibrated using the same lower bits stages in the extra pipeline branch, hence their performance becomes identical. This helps to compensate the mismatch among the branches. System wise, the I/Q mismatch can also be alleviated through this scheme. Since the calibration can be conducted at a much lower frequency, once every msec in our case, the overhead in power consumption is insignificant.

The measured SNR for the ADC is 60dB for the 5.5MHz 802.11b signal sampling at 44MS/s without disabling the last pipeline stage, and 64dB for the 550kHz BT signal at a sampling rate of 11MS/s. Fig. 6.29 shows the measured SNR in both receiving modes through a 14-bit DAC. The ADC power consumption are 20.2mW and 14.8mW for the 801.11b and BT receiving modes, respectively.

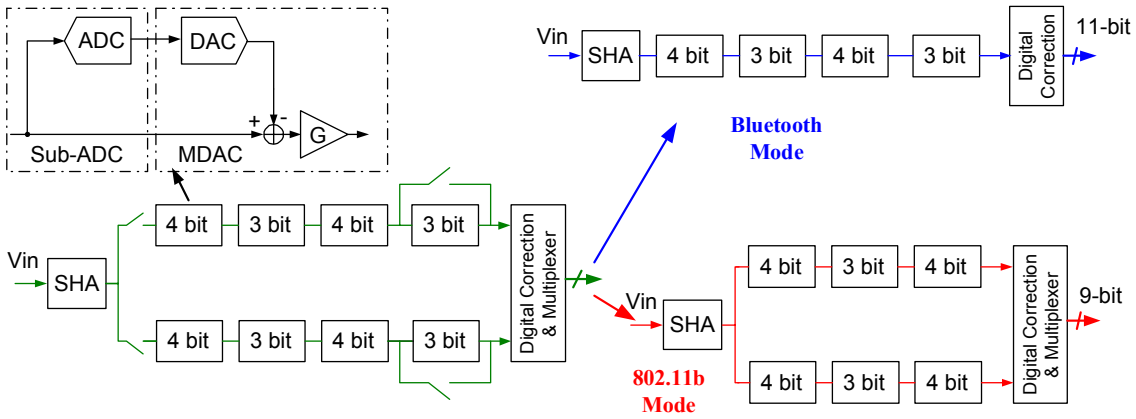


Fig. 6.27 Time-interleaved pipeline ADC architecture

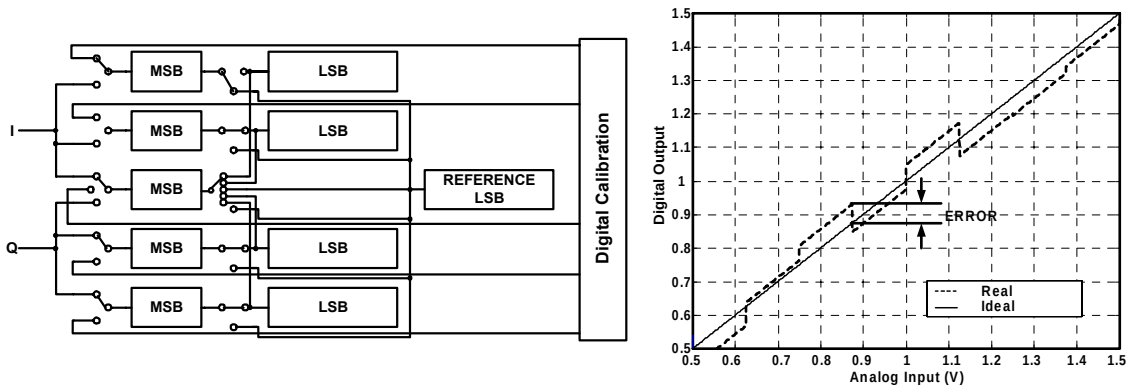
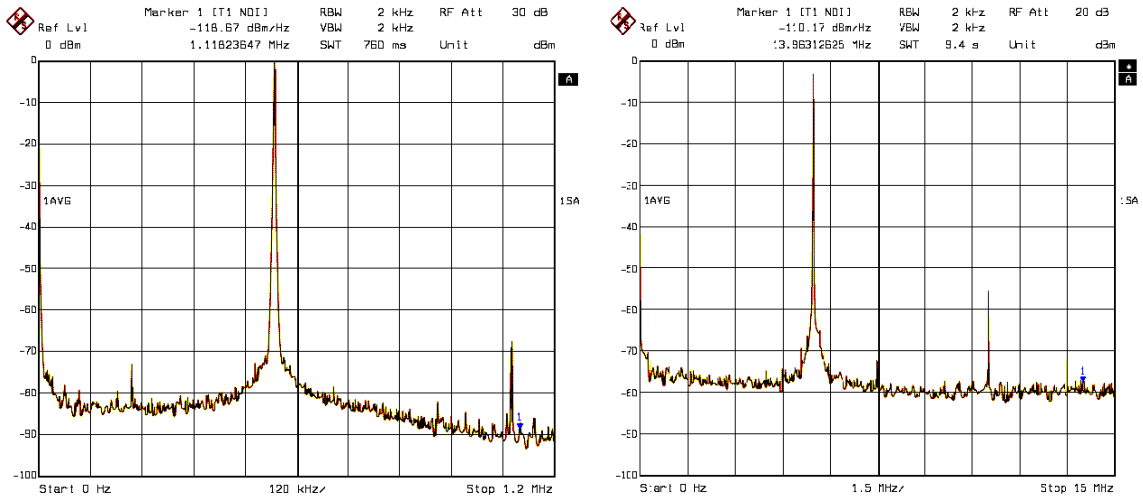


Fig. 6.28 Online digital calibration



(a) (b)

Fig. 6.29 Measured SNR of the ADC in (a) Bluetooth mode (64dB at 11Msample/s) (b) WiFi mode (48dB at 44Msample/s)

CHAPTER VII

CHAMELEON: EXPERIMENTAL RESULTS

7.1. System Chip

The Chameleon dual-mode Bluetooth/Wi-Fi receiver was fabricated in IBM 0.25μm BiCMOS technology. The total chip area including pads is 21mm^2. The total active area including pads and excluding unused space is 17.7mm^2. Details about the areas of the receiver blocks are in table 7.1. Note that the ADC is occupying about 58% of the receiver active area. The die micrograph of the chip is shown in Fig. 7.1. The chip was tested in a 4.5in×4.5in printed circuit board (PCB). The PCB was fabricated through PCB EXPRESS. The PCB uses FR-4 material with thickness 0.031″ in order to have reasonable track widths for impedance matching at 2.4GHz. Fig. 7.2 shows the simplified block diagram of testing board. Photograph of the actual PCB is depicted in Fig. 7.3. It measures 5.6mm×3.8mm.

Table 7.1 Receiver area breakdown

	area (mm^2)	area (%)
LNA+Mixer	0.5	4.5
PLL	1.6	13.2
Filter	1.9	15.9
VGA	1.0	8.1
ADC	7.0	58.3
Total active area	12.1	100.0
pads area	5.6	
active area+pads	17.7	
unused space	3.3	
Total active area	21.0	

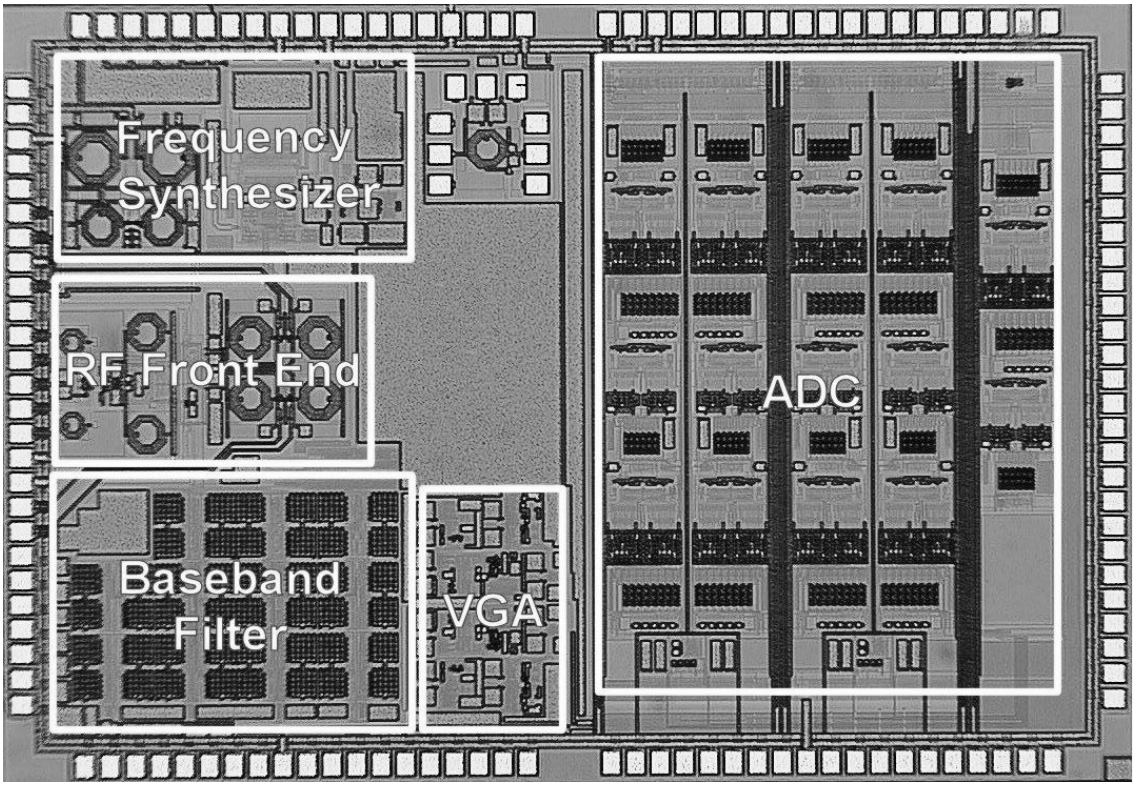


Fig. 7.1 Chameleon die micrograph

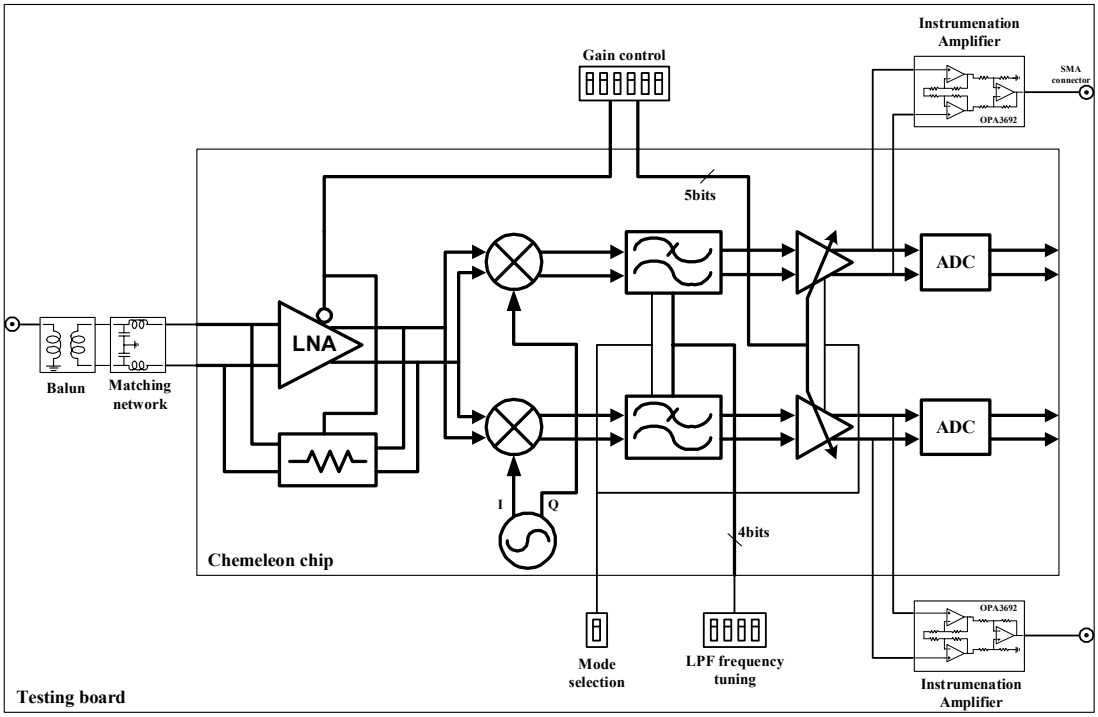


Fig. 7.2 Simplified block diagram of the testing board for the Chameleon chip

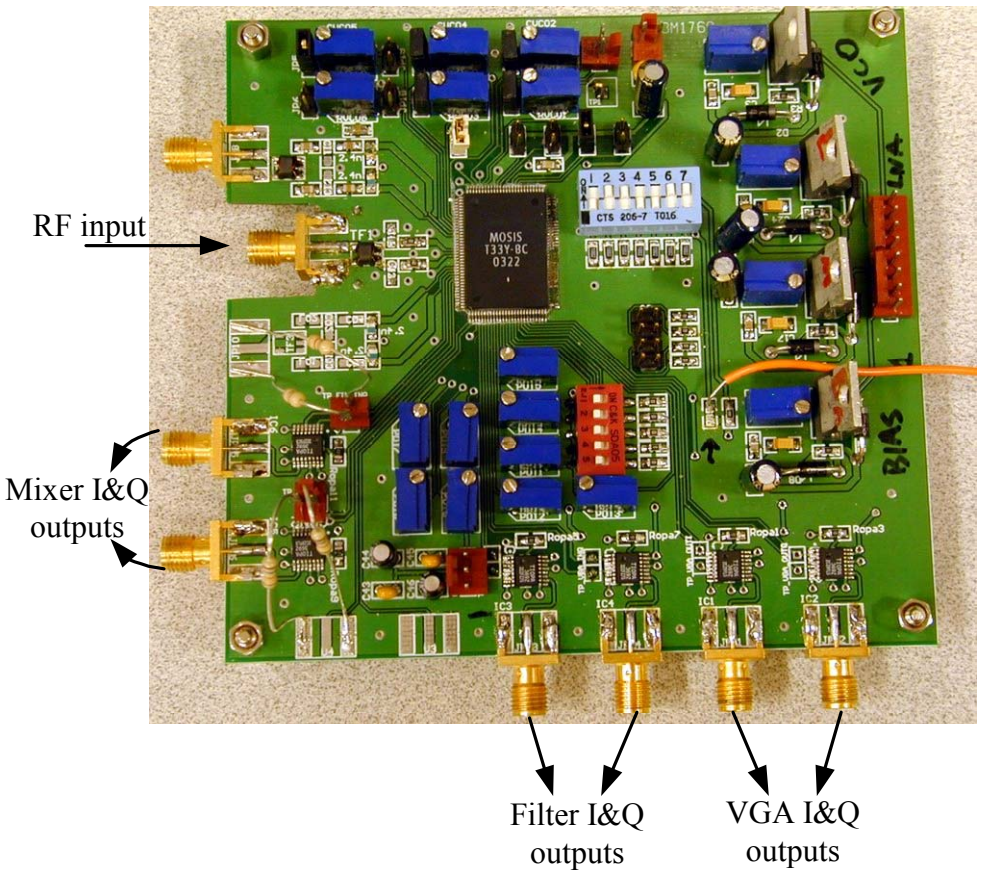


Fig. 7.3 Photograph of the PCB used to test the Chameleon chip

7.2. Experimental Results

7.2.1. BER vs. Input Power (Sensitivity)

Testing setup for sensitivity measurement is shown in Fig. 7.4. The sensitivity is measured as follows: the receiver gain is set to maximum and the LNA input is terminated by a 50Ω resistance. The integrated output noise (N_o) of the VGA is measured using the vector analyzer HP89410A. Then the signal is applied from the signal generator Agilent E44320 to the LNA input. The generator output signal level is set such that the signal level at the VGA output reaches $N_o + SNR_{\min}$. At this point, the signal

generator output power is the sensitivity level measured at the output of the generator. The corresponding signal level at the input of the LNA is given by:

$$\text{power at LNA input} = \text{generator power} - \text{cable/board attenuation} \quad (7.1)$$

The cable/board attenuation is estimated by using a similar board, with the same components values, and measuring the attenuation from the generator to the spectrum analyzer.

In order to obtain the BER vs. input power curve, the generator output level is swept from about 2 or 3 dBs below the sensitivity level to about 6dB above. For every input level, the following steps are done:

(1) Adjust the receiver gain such that the signal at the VGA output is within the ADC dynamic range.

(2) Measure the signal level at the VGA output

(3) The integrated noise at the VGA output measured by turning signal generator off and terminating the LNA input with a 50Ω resistance.

(4) At the VGA output, calculate the output SNR = signal level – noise level.

(5) Obtain BER from the simulated BER-SNR curve. Take measured receiver non-idealities into account.

Using the above technique, the BER vs. input power curves are obtained for Bluetooth and Wi-Fi modes. Fig. 7.5 shows the receiver BER versus the input signal power for Bluetooth mode and 802.11b mode at 11 and 5.5Mbps. The measured sensitivities are –86dBm for 802.11b 11Mbps (for BER = 10^{-5}) and –91dBm for Bluetooth (for BER = 10^{-3}).

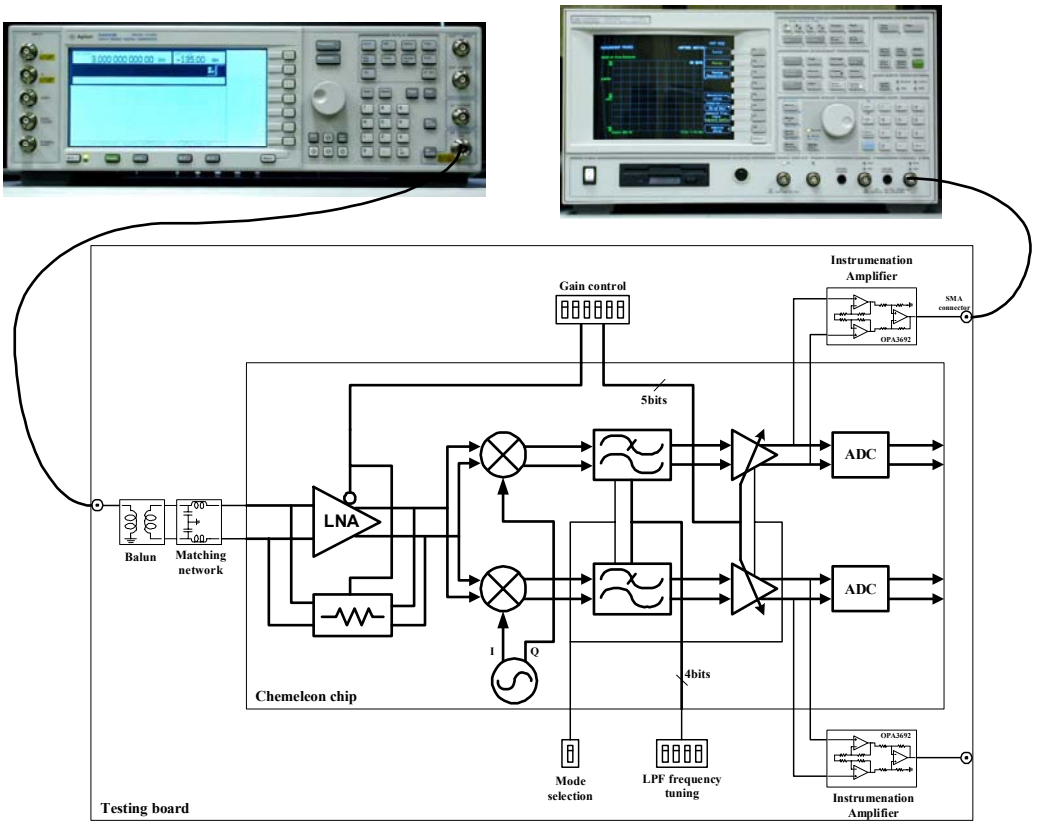


Fig. 7.4 Testing setup for NF measurement

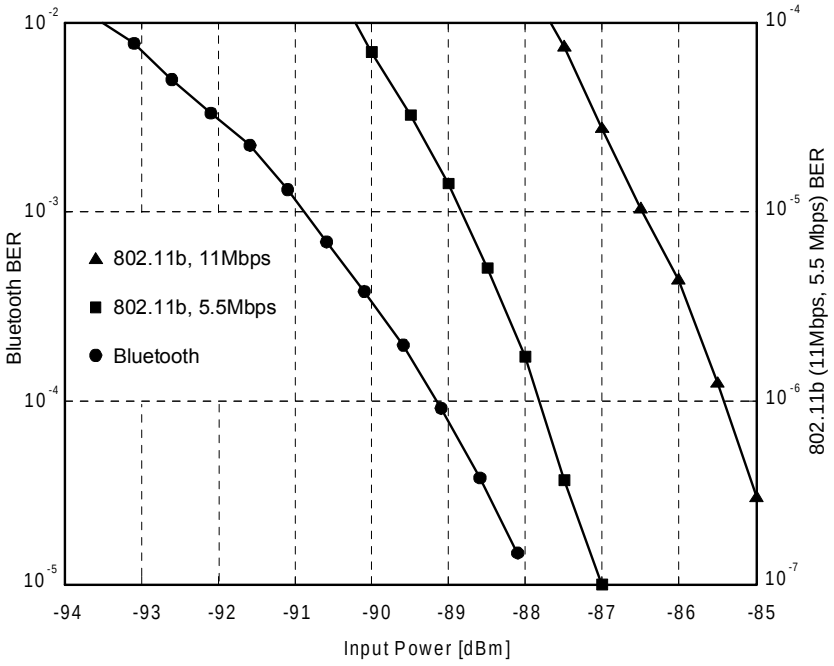


Fig. 7.5 BER performance of the dual-mode receiver

7.2.2. IIP3

Testing setup for IIP3 measurements is shown in Fig. 7.6. Two signal generators and a power combiner are used to run the IIP3 two-tone test. The test is done by sweeping the powers of two generators simultaneously and measuring the output fundamental and IM3 for every input power level. Since the two-tone test is usually done for out-of-band signals, the fundamental tones are attenuated by the filter, affecting the IIP3 measurement accuracy. Therefore, the fundamental power is measured when the input signal is in-band.

To summarize, the IIP3 measurement is done by performing the following steps for every generator power level:

- Set the two generators at the same power level.
- Set the frequency of the two generator frequencies, f_1 and f_2, according to the specified two-tone test.
- Measure the IM3 component at $2f_2 - f_1$ (f_2 is closer to the LO than f_1) at the VGA output.
- Turn off the one of the generators and set the frequency of the other one to an in-band frequency.
- Measure the fundamental component at the VGA output.

The above procedure is used to measure the IIP3 of the receiver. The receiver two-tone IIP3 curve in Wi-Fi mode is plotted in Fig. 7.7. The tones are applied at 12 and 25MHz from the LO frequency, when the LNA is in the high gain mode. Measured receiver IIP3 is –13dBm.

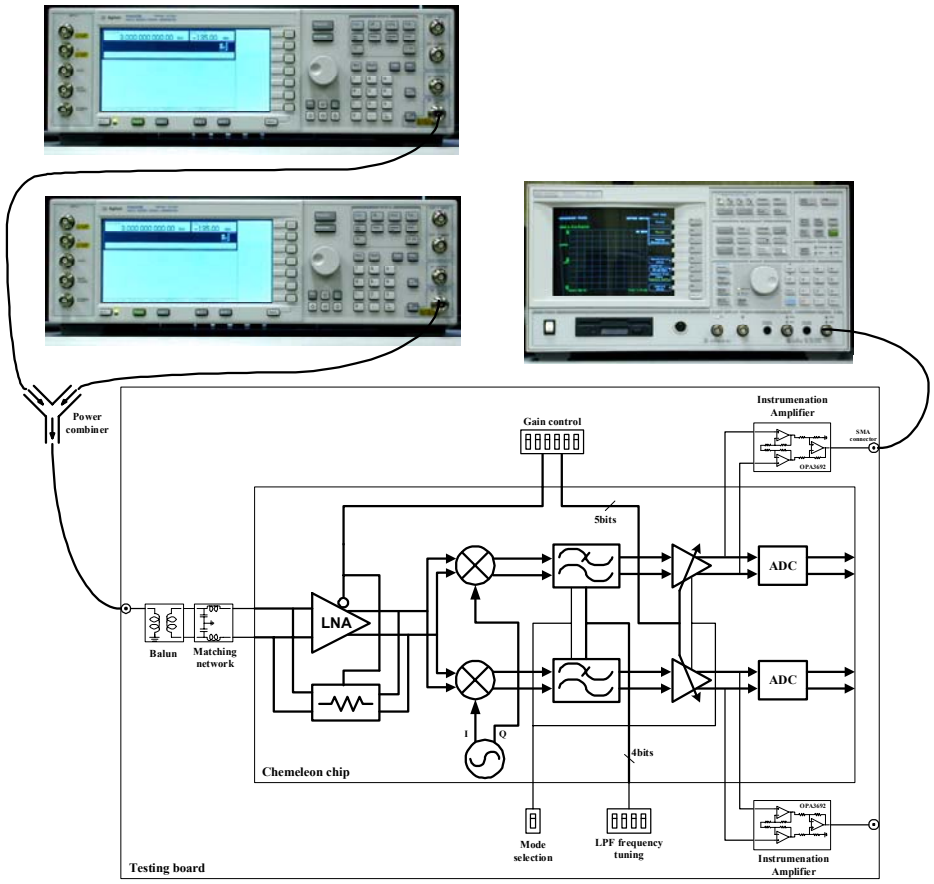


Fig. 7.6 Testing setup for IIP3/IIP2 two-tone test

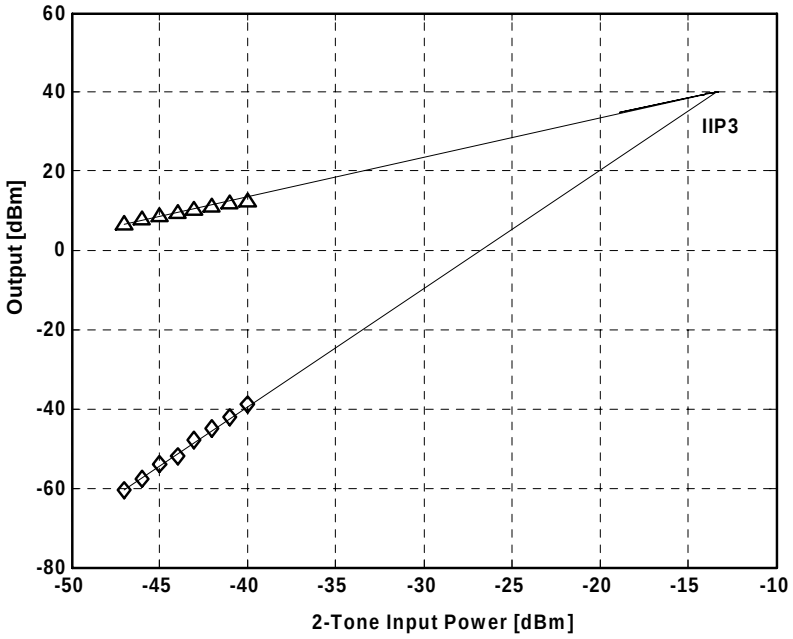


Fig. 7.7 Receiver IIP3 when LNA is in high gain mode (VGA gain = 12dB)

7.2.3. IIP2

The same connection setup shown in Fig. 7.6 is used for IIP2 two-tone test. The two tones from the signal generators are applied at 12 and 13MHz away from the LO frequency. The IIP2 is 10dBm and is plotted in Fig. 7.8. IIP2 tones are applied at 12 and 13MHz from the LO.

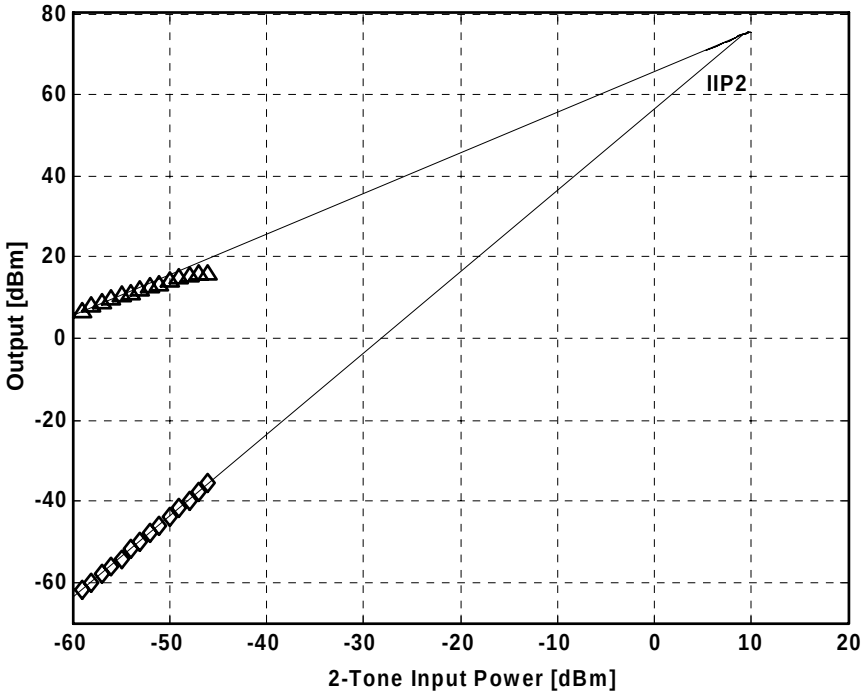


Fig. 7.8 Receiver IIP2 when LNA is in high gain mode (VGA gain = 24dB)

Table 7.2 presents a performance summary for this design and the two most similar previous reported implementations [37, 38]. Compared to previous implementations, our receiver consumes less power, has better Bluetooth sensitivity and comparable Wi-Fi sensitivity. Power consumption contributions of the receiver blocks is reported in Table 7.3. The temperature stability of the receiver performance is improved by a PTAT on-

chip bias of the RF-Front end, and the on-chip calibration capability of all base-band blocks.

Table 7.2 Performance summary of the Chameleon receiver

	[37]		[38]		This design	
	BT	WiFi	BT	WiFi	BT	WiFi
Receiver Architecture	Low-IF	Direct Conversion	Low-IF	Direct Conversion	Direct Conversion	Direct Conversion
Offset cancellation	Programmable loop		Injection at AGC input		ac coupling	
Channel select filter	separate		programmable		programmable	
Baseband amplifier	separate		shared		shared	
ADC	Not included		Not included		Included	
Filter bandwidth	1MHz (BPF)	7.5MHz (LPF)	1MHz (BPF)	7.5MHz (LPF)	600kHz (LPF)	6MHz (LPF)
Sensitivity	-82dbm	-88dBm	-80dBm	-92dBm (0dB SNR)	-91dBm	-86dBm
Technology	0.35μm CMOS		0.18μm CMOS		0.25μm BiCMOS	
Rx active current	46mA	65mA	60mA		41.3mA (w/ ADC)	45.6mA (w/ ADC)
ADC active current	-	-	-	-	13.4mA	15.6mA
IIP3	-7dBm	-8dBm	-12dBm		-13dBm	
IIP2	N/A	N/A	20dBm		10dBm	
Rx area (w/ pads)	N/A		16mm^2 (transceiver)		9mm^2 (w/o ADC)	
ADC area (w/ pads)	-	-	-	-	10mm^2	
Supply voltage	2.7V		1.8V		2.5V	

Table 7.3 Power consumption contribution of the receiver blocks

Block	Current drain in mA		power consumption (in %)	
	Wi-Fi	BT	Wi-Fi	BT
LNA/Mixer	13.6		29.2	32.9
PLL	12.5		26.9	30.3
Filter	2.7	0.9	5.8	2.2
VGA	2.1	0.9	4.5	2.2
ADC	15.6	13.4	33.5	32.4
Total	46.5	41.3	100.0	100.0

CHAPTER VIII

CONCLUSIONS

The dissertation has examined in detail a number of issues related to wireless radio design both at the system and at the building block level. At the system level, a low-IF receiver design for Bluetooth standard has been presented. The receiver was implemented in TSMC 0.35μm CMOS process. This receiver yields comparable or better performance than most published commercial receivers that use more expensive advanced technologies.

To target wider range of wireless applications, a dual-mode 802.11b/Bluetooth receiver has been implemented in IBM 0.25μm BiCMOS process. Direct conversion architecture was used for both standards to achieve maximum level of integration and block sharing. Compared to recently reported dual-mode receivers, this receiver has a lower current consumption in both operating modes, better Bluetooth sensitivity, and comparable Wi-Fi sensitivity.

At the building block level, a pseudo differential OTA-C complex filter design for the low-IF Bluetooth receiver has been implemented. A pseudo differential OTA is used to comply with low voltage operation. To save area and power consumption, a sound scheme for common-mode control is implemented using a minimum number of CMFF and CMFB circuits. The filter uses a non-conventional frequency tuning circuit architecture that has less silicon area and design complexity compared to the conventional PLL.

To be used as frequency modulator in the Bluetooth transmitter, a ROM-less DDFS has been implemented in AMI 0.5μm CMOS process. The DDFS architecture is based on piecewise linear approximation of the sine function. The proposed architecture is shown to have significant area and power savings at high clock rates compared to the recently reported structures in the literature. This renders the proposed structure more suitable for low power portable applications.

As part of the dual-mode receiver, variable gain amplifier with offset cancellation technique has been implemented. Offset cancellation is done via low cutoff frequency HPFs in between the VGA stages.

In summary, the contribution highlights of this dissertation are:

- 100MHz Direct digital frequency synthesizer.
- Complex filter with frequency tuning for low-IF Bluetooth receiver.
- Dual mode 802.11/Bluetooth receiver design.
- 0-62dB variable gain amplifier with offset cancellation scheme.

REFERENCES

[1] *Specifications of the Bluetooth System, Version 1.0 B*, Overland Park, KS, Bluetooth SIG, 1999.

[2] R. Morrow, *Bluetooth Operation and Use*. New York: McGraw-Hill, 2002.

[3] *Specifications of the IEEE 802.11b System*, New York: IEEE, 1999.

[4] J. Ross, *The Book of Wi-Fi*. San Francisco: No Starch Press, 2003.

[5] C. Andren, K. Halford, and M. Webster, "CCK, the New IEEE 802.11 Standard for 2.4GHz Wireless LAN," in *International IC*, Taipei, Taiwan, pp. 25-39, 2001.

[6] J. Lansford, A. Stephens, and R. Nevo, "Wi-Fi (802.11b) and Bluetooth Enabling Coexistence," *IEEE Network*, vol. 15, pp. 20-27, Sept.-Oct. 2001.

[7] G. F. Ross, "The Evolution of UWB Technology," in *Radio and Wireless Conf.,* Boston, Massachusetts, pp. 1-4, Aug. 2003.

[8] "Revision of Part 15 of the Commission's Rules Regarding Ultra-wideband Transmission Systems,," in *ET-Docket 98-153, FCC, http://www.fcc.gov/oet/ dockets/et98-153/*, accessed on 10/26/2003.

[9] J. D. Taylor, *Introduction to Ultra-Wideband Radar Systems*. Boca Raton, FL: CRC Press, 1995.

[10] M. Rofheart, "XtremeSpectrum Multimedia WPAN PHY," in *IEEE 802.15.3 Working Group Submission,*, La Jolla, CA: IEEE, July 2000.

[11] B. Razavi, "Architectures and Circuits for RF CMOS receivers," in *Proc. IEEE Custom Integrated Circuits Conf.*, pp. 393-400, May 1998.

[12] B. Razavi, "Design Considerations for Direct-Conversion Receivers," *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 428-435, June 1997.

[13] J. Crols and M. S. J. Steyaert, "Low-IF Topologies for High-performance Analog Front Ends of Fully Integrated Receivers," *IEEE Trans. Circuits Syst. II*, vol. 45, pp. 269-283, Mar. 1998.

[14] K. S. Shanmugan, *Digital and Analog Communication Systems*. New York: Wiley, 1979.

[15] W. Sheng, B. Xia, A. E. Emira, C. Xin, A. Y. Valero-Lopez, S. T. Moon, and E. Sánchez-Sinencio, "A 3-V, 0.35μm CMOS Bluetooth Receiver IC," *IEEE J. of Solid-State Circuits*, vol. 38, pp. 30-42, Jan. 2003.

[16] B. Razavi, *RF Microelectronics*. Upper Saddle River, NJ: Prentice Hall, 1998.

[17] K. Philips, "A 4.4mW 76dB Complex ΣΔ ADC for Bluetooth Receivers," in *IEEE Int. Solid State Circuits Conf.*, San Francisco, Feb. 2003.

[18] F. Behbahani, Y. Kishigami, J. Leete, and A. A. Abidi, "CMOS Mixers and Polyphase Filters for Large Image Rejection," *IEEE J. of Solid-State Circuits*, vol. 36, pp. 873-887, June 2001.

[19] B. Xia, C. Xin, W. Sheng, A. Y. Valero-lopez, and E. Sánchez-Sinencio, "A GFSK Demodulator for Low-IF Bluetooth Receiver," *IEEE J. of Solid-State Circuits*, vol. 38, pp. 1397-1400, Aug. 2003.

[20] P. Andreani, S. Mattisson, and B. Essink, "A CMOS gm-C Polyphase Filter with High Image Band Rejection," in *26th European Solid-State Circuits Conf.*, Stockholm, Sweden, Sep. 2000.

[21] J. Crols and M. Steyaert, "An Analog Integrated Polyphase Filter for a High Performance Low-IF Receiver," in *Symp. VLSI Circuits Dig. Tech. Papers*, Kyoto, Japan, pp. 87-88, June 1995.

[22] F. Behbahani, H. Firouzkouhi, R. Chokkalingam, S. Delshadpour, A. Kheirkhahi, M. Nariman, M. Conta, and S. Bhatia, "A Fully Integrated Low-IF CMOS GPS Radio with On-Chip Analog Image Rejection," *IEEE J. Solid-State Circuits*, vol. 37, pp. 1721-1727, Dec. 2002.

[23] P. T. M. van Zeijl, J.-W. Eikenbroek, P.-P. Vervoot, S. Setty, J. Tangenberg, G. Shipton, E. Kooistra, I. Keekstra, and D. Belot, "A Bluetooth Radio in 0.18 µm CMOS," in *IEEE Int. Solid State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, pp. 86-87, Feb. 2002.

[24] A. S. Sedra, W. M. Snelgrove, and R. Allen, "Analogue Bandpass Filters Design by Linearly Shifting Real Low-Pass Prototypes," in *Proc. IEEE Int. Symp. Circuits and Systems*, Kyoto, Japan, pp. 1223-1226, 1985.

[25] S. A. Jantzi, K.W. Martin, and A. S. Sedra, "The Effects of Mismatch in Complex Bandpass ΣΔ Modulators," in *Proc. IEEE Int. Symp. Circuits and Systems*, pp 227-230, May 1996.

[26] B. Nauta and E. Seevinck, "Linear CMOS Transconductance Element for VHF Filters," *Electronics Letters*, vol. 25, pp. 448-450, Mar. 1989.

[27] D. A. Sunderland, R.A. Strauch, S.S. Wharfield, H.T. Peterson, and C. R. Cole, "CMOS/SOS Frequency Synthesizer LSI Circuit for Spread Spectrum Communications," *IEEE J. of Solid-State Circuits*, vol. 19, pp. 497-505, Aug. 1984.

[28] H. T. Nicholas, H. Samueli, and B. Kim, "The Optimization of Direct Digital Frequency Synthesizer Performance in the Presence of Finite Word Length Effects," in *Proc. 42th Annual Frequency Control Symp.*, Baltimore, MD, pp. 357-363, June 1988.

[29] A. M. Sodagar and G. R. Lahiji, "Mapping from Phase to Sine-amplitude in Direct Digital Frequency Synthesizers Using Parabolic Approximation," *IEEE Trans Circuits Syst.*, vol. 47, pp. 1452-1457, Dec. 2000.

[30] A. Bellaouar, M.S. O'brecht, A.M. Fahim, and M. I. Elmasry, "Low-power Direct Digital Frequency Synthesis for Wireless Communications," *IEEE J. of Solid-State Circuits*, vol. 35, pp. 385-390, Mar. 2000.

[31] S. Mortezapour and E. K. F. Lee, "Design of Low-power ROM-less Direct Digital Frequency Synthesizer Using Nonlinear Digital-to-Analog Converter," *IEEE J. of Solid-State Circuits*, vol. 34, pp. 1350-1359, Oct. 1999.

[32] J. Jiang and E. K. F. Lee, "A ROM-less Direct Digital Frequency Synthesizer Using Segmented Nonlinear Digital-to-Analog Converter," in *IEEE Custom Integrated Circuits Conf.*, San Diego, CA, pp. 165-168, May 2001.

[33] A. N. Mohieldin, A. Emira, and E. Sánchez-Sinencio, "A 100MHz, 8mW ROM-less Quadrature Direct Digital Frequency Synthesizer," *IEEE J. of Solid-State Circuits*, vol. 37, pp. 1235-1243, Oct. 2002.

[34] C. Meenakarn and A. Thanachayanont, "100-MHZ CMOS Direct Digital Synthesizer with 10-bit DAC," in *Proc. Asia-Pacific Conf. Circuits and Syst.*, Bangkok, Thailand, pp. 385-388, Oct. 2002.

[35] B. Provost and E. Sánchez-Sinencio, "On-chip Ramp Generators for Mixed-signal BIST and ADC Self-test," *IEEE J. of Solid-State Circuits*, vol. 38, pp. 263-273, Feb. 2003.

[36] D. A. Johns and K. Martin, *Analog Integrated Circuit Design*. New York: John Wiley & Sons, 1997.

[37] H. Darabi, J. Chiu, S. Khorram, H. Kim, Z. Zhou, E. Lin, S. Jiang, K. Evans, E. Chien, B. Ibrahim, E. Geronaga, L. Tran, and R. Rofougaran, "A Dual Mode 802.11b/Bluetooth Radio in 0.35μm CMOS," in *IEEE Int. Solid State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, pp. 86-87, Feb. 2003.

[38] T. Cho, D. Kang, S. Dow, C. H. Heng, and B. S. Song, "A 2.4Ghz Dual-Mode 0.18μm CMOS Transceiver for Bluetooth and 802.11b," in *IEEE Int. Solid State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, pp. 88-89, Feb. 2003.

[39] E. Chien, W. Feng, Y. A. Hsu, and L. Tse, "A 2.4GHz CMOS Transceiver and Baseband Processor Chipset for 802.11b Wireless LAN Application," in *IEEE Int.*

Solid State Circuits Conf. (ISSCC) Dig. Tech. Papers, San Francisco, CA, pp. 1-10, Feb. 2003.

[40] P. M. Stroet, R. Mohindra, S. Hahn, A. Schuur, and E. Riou, "A Zero-IF Single-chip Transceiver for Up to 22 Mb/s QPSK 802.11b Wireless LAN," in *IEEE Int. Solid State Circuits Conf. (ISSCC) Dig. Tech. Papers*, pp. 204-205, Feb. 2001.

[41] K. Y. Lee, S. W. Lee, Y. Koo, H. K. Huh, H. Y. Nam, J. W. Lee, J. Park, K. Lee, D. K. Jeong, and W. Kim, "Full-CMOS 2-GHz WCDMA Direct Conversion Transmitter and Receiver," *IEEE J. of Solid-State Circuits*, vol. 38, pp. 43-53, Jan. 2003.

[42] T. Arai and T. Itakura, "A Baseband Gain-controlled Amplifier with a Linear-in-dB Gain Range from 14dB to 76dB and a Fixed Corner Frequency DC Offset Canceler," in *IEEE Int. Solid State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, pp. 136-137, Feb. 2003.

[43] A. Savla, A. Ravindran, J. Leonard, and M. Ismail, "System Analysis of a Multi-Standard Direct Conversion Wireless Receiver," in *45th Midwest Symp. on Circuits and Systems*, Tulsa, OK, vol. 3, pp. 401-404, Aug. 2002.

[44] H. Darabi, S. Khorram, H.-M. Chien, M.-A. Pan, S. Wu, S. Moloudi, J. C. Leete, J. J. Rael, M. Syed, R. Lee, B. Ibrahim, M. Rofougaran, and A. Rofougaran, "A 2.4-GHz CMOS Transceiver for Bluetooth," *IEEE J. Solid-State Circuits*, vol. 36, pp. 2016-2024, Dec. 2001.

[45] R. Adler, "A Study of Locking Phenomena in Oscillators," *Proc. IRE*, vol. 34, pp. 351-357, June 1946.

[46] K. Kurokawa, "Injection Locking of Microwave Solid-State Oscillators," *Proc. IEEE*, vol. 61, pp. 1386-1410, Oct. 1973.

[47] P. T. M. v. Zeijl, J.-W. Eikenbroek, P.-P. Vervoot, S. Setty, J. Tangenberg, G. Shipton, E. Kooistra, I. Keekstra, and D. Belot, "A Bluetooth Radio in 0.18 µm CMOS," in *IEEE Int. Solid State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, pp. 86-87, Feb. 2002.

[48] F. O. Eynde, J.-J. Schmit, V. Charlier, R. Alexandre, C. Sturman, K. Coffin, B. Mollekens, J. Craninckx, S. Terrijn, A. Monterastelli, S. Beerens, P. Goetschalckx, M. Ingels, D. Joos, S. Guncer, and A. Pontioglu, "A Fully-Integrated Single-chip SOC for Bluetooth," in *IEEE Int. Solid State Circuits Conf. (ISSCC) Dig. Tech. Papers*, pp. 196-197, Feb. 2001.

[49] M. Chen, K. H. Wang, D. Zhao, L. Dai, Z. Soe, and P. Rogers, "A CMOS Bluetooth Radio Transceiver Using a Sliding-IF Architecture," in *IEEE Custom Integrated Circuits Conf.*, San Jose, CA, Sept. 2003.

[50] H. T. Friis, "Noise Figure of Radio Receivers," *Proc. IRE*, vol. 32, pp. 419-422, July 1944.

[51] D. K. Shaeffer and T. H. Lee, "A 1.5-V, 1.5-GHz CMOS Low Noise Amplifier," *IEEE Journal of Solid-State Circuits*, vol. 32, pp. 745-759, May 1997.

[52] W. Rhee, "Design of High-performance CMOS Charge Pumps in Phase-locked Loops," in *IEEE International Symposium on Circuits and Systems*, Orlando, FL, vol. 2, pp. 545-548, June 1999.

[53] S. Solis-Bustos, J. Silva-Martínez, F. Maloberti, and E. Sánchez-Sinencio, "A 60-dB Dynamic Range CMOS Sixth-order 2.4-Hz Low-pass Filter for Medical Applications," *IEEE Trans Circuits Syst. II*, vol. 47, pp. 1391-1398, Dec. 200.

[54] K. Shu, E. Sánchez-Sinencio, J. Silva-Martínez, and S. J. K. Embabi, "A 2.4-GHz Monolithic Fractional-N Frequency Synthesizer with Robust Phase-Switching Prescaler and Loop Capacitance Multiplier," *IEEE J. of Solid-State Circuits*, vol. 38, pp. 866-874, June 2003.

[55] T. Itakura, T. Ueno, H. Tanimoto, A. Yasuda, R. Fujimoto, T. Arai, and H. Kokatsu, "A 2.7-V, 200-kHz, 49-dBm, Stopband-IIP3, Low-noise, Fully Balanced g_m-C Filter IC," *IEEE J. Solid-State Circuits*, vol. 34, pp. 1155-1159, Aug. 1999.

[56] A. Motamed, C. Hwang, and M. Ismail, "A Low-Voltage Low-Power Wide-Range CMOS Variable Gain Amplifier," *IEEE Trans Circuits Syst.*, vol. 45, pp. 800-811, July 1998.

[57] M. Mostafa, H. Elwan, A. Bellaour, B. Kramer, and S. Embabi, "A 110MHz 70dB CMOS Variable Gain Amplifier," in *IEEE International Symposium on Circuits and Systems*, Orlando, FL, vol. 2, pp. 623-631, July 1999.

[58] H. Elwan, A. El-Adawi, M. Ismail, H. K. Olsson, and A. Soliman, "Digitally Controlled dB-Linear CMOS Variable Gain Amplifier," *Electronics Letters*, vol. 35, pp. 1725-1727, Sept. 1999.

[59] H. O. Elwan and M. Ismail, "Low Voltage Digitally Controlled dB-Linear CMOS VGA Circuit for Wireless Communication," in *IEEE International Conference on Electronics, Circuits, and Systems*, Lisboa, Portugal, pp. 341-344, Sept. 1998.

[60] J. J. F. Rijns, "CMOS Low-Distortion High-Frequency Variable-Gain Amplifier," *IEEE J. of Solid-State Circuits*, vol. 31, pp. 1029-1034, July 1996.

[61] G. Gomez and B. Haroun, "A 1.5 V 2.4/2.9 mW 79/50 dB DR ΣΔ Modulator for GSM/WCDMA in a 0.13μm Digital Process," in *IEEE Int. Solid State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, pp. 306-307, Feb. 2002.

[62] S.-H. Lee and B.-S. Song, "Digital-domain Calibration of Multistep Analog-to-Digital Converters," *IEEE Journal of Solid-State Circuits*, vol. 27, pp. 1679-1688, Dec. 1992.

APPENDIX A

MATLAB CODES FOR CALCULATING THE OPTIMUM SLOPE VALUES FOR THE DDFS

```
% This code calculates the optimum slope values for
% specific phase resolution, number of segments, and
% normalized resistance step
global slope_res;
global phase_res;
global amp_res;
global amp;
global phase_step
global seg_length;
slope_res=8;
phase_res=5;
amp_res=8;
amp=1;
Nseg=8;
seg_length=2^(phase_res+1)/Nseg;
phase_step=pi/2/2^(phase_res+1);
phase=linspace(0,pi/2,2^(phase_res+1)+1);
ideal_sin=amp*sin(phase);
offsety(1)=0;
for i=1:Nseg
  ideal_curve=ideal_sin((i-1)*seg_length+1:i*seg_length+1)-
           ideal_sin((i-1)*seg_length+1);
  [segment(i,:) slope(i) yend(i)]=
      find_fixed_segment(offsety(i), ideal_curve);
offsety(i+1)=yend(i)-ideal_curve(end);
end
real_curve(1:seg_length)=slope(1)*linspace(0.5,seg_length
                  -0.5,seg_length)*phase_step;
for j=2:Nseg
  for i=1:seg_length
    real_curve((j-1)*seg_length+i)=sum(slope(1:j-1))*
     seg_length*phase_step+slope(j)*(i-0.5)*phase_step;
  end
end
real_phase=linspace(0,pi/2-pi/2/2^(phase_res+1),
2^(phase_res+1))+pi/2/2/2^(phase_res+1);
```

```
% This function calculates optimum slope of 1st segment
function [segment, slope, yend]=
                find_first_segment(offsety, curve)
global slope_res;
global phase_res;
global amp_res;
global amp;
global phase_step
global seg_length

slope=2^slope_res/amp;
segment=slope*(0:seg_length)*phase_step;
max_error=max(abs(curve-segment));
while max_error>=max(abs(curve-segment))
  max_error=max(abs(curve-segment));
  slope=1/(1/slope+amp/2^slope_res);
  segment=offsety+slope*(0:seg_length)*phase_step;
endif
slope ~=0
  slope=slope-amp/2^slope_res;
end
yend=offsety+slope*seg_length*phase_step;
```

```
% This function calculates the optimum slope in a
% specific segment

function [segment, slope, yend]=find_fixed_segment(offsety, curve)
global slope_res;
global phase_res;
global amp_res;
global amp;
global phase_step
global seg_length

slope=2^slope_res/amp;
segment=offsety+slope*(0:seg_length)*phase_step;
max_error=max(abs(curve-segment));
while max_error>=max(abs(curve-segment))
  max_error=max(abs(curve-segment));
  slope=1/(1/slope+amp/2^slope_res);
  segment=offsety+slope*(0:seg_length)*phase_step;
end

yend=offsety+slope*seg_length*phase_step;
```

APPENDIX B

MATLAB CODES FOR GFSK DEMODULATOR

```
%%% This code is to calculate the BER for GFSK demodulator
clear all;
j=sqrt(-1);
% define non-idealities
samp=10;
whp=1e3*2/samp/1e6;
bwlp=600e3*2/samp/1e6;
dcoff=0.1;
deltaphi=5/180*pi;
deltamag=0.1;
Qbit=5;
delay=9;
EbN=13;
Num_data=50000;
% Generate random data
data=2*round(rand(1, Num_data))-1;
smpl_sp=Num_data*samp;

sp_data=zeros(1, Num_data*samp);
for i=1 : Num_data
        sp_data((i-1)*samp+1:i*samp)=data(i);
end

% Add Guassian pulse shape
BT=.5;
t=[-samp:1:samp];
ht=sqrt(pi*2/log(2))*BT/samp*exp(-(pi^2*2*(BT/samp)^2/log(2))*power(t,2));
psh_sig=conv(sp_data, ht);


% BT/samp=B, the Guassin filter cause 1 bit delay
% Define complex envelope of the GFSK signal
ind=0.32; % norminal value, modulation index for BT is 0.28-0.35
phi=0;
gfsk=zeros(1, smpl_sp-2*samp);
for i=1 : smpl_sp-2*samp,
        phi=phi+psh_sig(i+2*samp); % Get rid of the first 2 bits to obtain
                                   % better synchronization
        gfsk(i)=cos(ind*pi*phi/samp+deltaphi/2)*(1+deltamag/2)+
         j*sin(ind*pi*phi/samp-deltaphi/2)*(1-deltamag/2);
end
```

```
% Match filters
d00=ones(1, 4*samp);
d01=[ones(1, 3*samp) (-1)*ones(1,samp)];
d10=[ones(1, 3*samp)*(-1) ones(1,samp)];
d11=ones(1, 4*samp)*(-1);
for i=1:samp,
  d00(i+samp)=-1;
  d00(i+samp*2)=-1;
  d01(i+samp)=-1;
  d01(i+samp*2)=1;
  d10(i+samp)=1;
  d10(i+samp*2)=-1;
        d11(i+samp)=1;
  d11(i+samp*2)=1;
end
sd00=conv(d00, ht);
sd01=conv(d01, ht);
sd10=conv(d10, ht);
sd11=conv(d11, ht);
s00=sd00(2*samp+1: samp*4);
s01=sd01(2*samp+1: samp*4);
s10=sd10(2*samp+1: samp*4);
s11=sd11(2*samp+1: samp*4);
phi00=0;
phi01=0;
phi10=0;
phi11=0;
for i=1 : 2*samp,
  phi00=phi00+s00(i);
  mf00(i)=((exp(j*ind*pi*phi00/samp))').';
  phi01=phi01+s01(i);
  mf01(i)=((exp(j*ind*pi*phi01/samp))').';
  phi10=phi10+s10(i);
  mf10(i)=((exp(j*ind*pi*phi10/samp))').';
  phi11=phi11+s11(i);
  mf11(i)=((exp(j*ind*pi*phi11/samp))').';
end
d000=[ones(1, samp) -ones(1, 3*samp) ones(1, samp)];
d001=[ones(1, samp) -ones(1, 2*samp) ones(1, samp) -ones(1, samp)];
d010=[ones(1, samp) -ones(1, samp) ones(1, samp) -ones(1, samp) ones(1, samp)];
d011=[ones(1, samp) -ones(1, samp) ones(1, 2*samp) ];
d100=[-ones(1, samp) ones(1, samp) -ones(1, samp) -ones(1, samp) ones(1, samp)];
d101=[-ones(1, samp) ones(1, samp) -ones(1, samp) ones(1, samp) -ones(1, samp)];
d110=[-ones(1, samp) ones(1, 2*samp) -ones(1, samp) ones(1, samp)];
d111=[-ones(1, samp) ones(1, 3*samp) -ones(1, samp)];
```

```
sd000=conv(d000, ht);
sd001=conv(d001, ht);
sd010=conv(d010, ht);
sd011=conv(d011, ht);
sd100=conv(d100, ht);
sd101=conv(d101, ht);
sd110=conv(d110, ht);
sd111=conv(d111, ht);
s000=sd000(2*samp+1: samp*5);
s001=sd001(2*samp+1: samp*5);
s010=sd010(2*samp+1: samp*5);
s011=sd011(2*samp+1: samp*5);
s100=sd100(2*samp+1: samp*5);
s101=sd101(2*samp+1: samp*5);
s110=sd110(2*samp+1: samp*5);
s111=sd111(2*samp+1: samp*5);
phi000=0;
phi001=0;
phi010=0;
phi011=0;
phi100=0;
phi101=0;
phi110=0;
phi111=0;
for i=1 : 3*samp,
  phi000=phi000+s000(i);
  mf000(i)=((exp(j*ind*pi*phi000/samp))').';
  phi001=phi001+s001(i);
  mf001(i)=((exp(j*ind*pi*phi001/samp))').';
  phi010=phi010+s010(i);
  mf010(i)=((exp(j*ind*pi*phi010/samp))').';
  phi011=phi101+s011(i);
  mf011(i)=((exp(j*ind*pi*phi011/samp))').';
  phi100=phi100+s100(i);
  mf100(i)=((exp(j*ind*pi*phi100/samp))').';
  phi101=phi101+s101(i);
  mf101(i)=((exp(j*ind*pi*phi101/samp))').';
  phi110=phi110+s110(i);
  mf110(i)=((exp(j*ind*pi*phi110/samp))').';
  phi111=phi111+s111(i);
  mf111(i)=((exp(j*ind*pi*phi111/samp))').';
end
```

```
% Add white noise, define Eb/N
%EbN=(8:0.5:16);
for k=1:length(EbN),
        sigma=sqrt(samp/2/power(10,(EbN(k)/10)));
        %sigma=0;
        Nn=length(gfsk);
        No_I=randn(1, Nn)*sigma;
        No_Q=randn(1, Nn)*sigma;

% LPF filters the noise
        order=5;
        [BBS,ABS]=butter(order, bwlp);
%       order=4;
%       [BBS,ABS]=cheby1(order, 0.5, bwlp);

% Filter the noise through LPF
  Nk_I=filter(BBS,ABS,No_I);
  Nk_Q=filter(BBS,ABS,No_Q);
  gfsk0=filter(BBS,ABS,gfsk);
%   gfsk0=gfsk;
% Let the signal, noise go through the HPF twice
  [BHP,AHP]=butter(1,whp, 'high');
  Nk_I1=filter(BHP,AHP,Nk_I);
  Nk_Q1=filter(BHP,AHP,Nk_Q);
  gfsk1=filter(BHP,AHP,gfsk0);

  Nk_I2=filter(BHP,AHP,Nk_I1);
  Nk_Q2=filter(BHP,AHP,Nk_Q1);
  gfsk2=filter(BHP,AHP,gfsk1);

% Received signal
        rx_sigo=gfsk2+(Nk_I+j*Nk_Q)+dcoff;
   rx_sig=[rx_sigo(delay+1:end) zeros(1,delay)];

% Quantization
   smpl_rx=length(rx_sig);
   Nrx=smpl_rx/samp;
   Ts=linspace(0, 1e-6*Nrx, smpl_rx)';
   RxReal=(real(rx_sig)).';
   RxImg=(imag(rx_sig)).';
   qStep=2^Qbit;
   qinv=4/qStep;
   DataIn=[RxReal RxImg];
```

```
%Let the received signal go through the quantizer
   SimStp=Ts(2)-Ts(1);
   options=simset('Reltol',1e-3,'Maxstep',SimStp);
   sim('quantizer', Ts(end), options);
   rx_sig=Iout.'+j*(Qout.');
% Run the received signal through match filter bank
% Calculate the first bit
         lambda(1)=abs(sum(rx_sig(1:2*samp).*mf00));
  lambda(2)=abs(sum(rx_sig(1:2*samp).*mf01));
  lambda(3)=abs(sum(rx_sig(1:2*samp).*mf10));
  lambda(4)=abs(sum(rx_sig(1:2*samp).*mf11));
  [mxl, ix]=max(lambda);
  Dr(1)=floor(ix/3)*2-1;

         for i=2:(length(rx_sig)/samp-1)
         icur=samp*(i-2);
         pSt=icur+1;
    pEd=icur+samp*3;
    if Dr(i-1)==1
                  lambda(1)=abs(sum(rx_sig(pSt:pEd).*mf100));
                  lambda(2)=abs(sum(rx_sig(pSt:pEd).*mf101));
                  lambda(3)=abs(sum(rx_sig(pSt:pEd).*mf110));
      lambda(4)=abs(sum(rx_sig(pSt:pEd).*mf111));
    else
      lambda(1)=abs(sum(rx_sig(pSt:pEd).*mf000));
                  lambda(2)=abs(sum(rx_sig(pSt:pEd).*mf001));
                  lambda(3)=abs(sum(rx_sig(pSt:pEd).*mf010));
      lambda(4)=abs(sum(rx_sig(pSt:pEd).*mf011));
                  end
         [mxl, ix]=max(lambda);
         Dr(i)=floor(ix/3)*2-1;
         end
   Dr=Dr(9:(end-8));
         Ds=data(10:Num_data-10);
         De=Dr-Ds;
         err=find(De);
         BER(k)=(length(err))/length(Ds)*100;
         [np f]=psd(Nk_I, 8192, samp*1e6);
         [sp f]=psd(real(gfsk2), 8192, samp*1e6);
  snr(k)=10*log10(sum(sp)/sum(np));
 end
BER
```

APPENDIX C

MATLAB CODES FOR CALCULATING EQUIVALENT IIP3, IIP2, AND LO PHASE NOISE IN WI-FI

```
%%% This code is to calculate the equivalent IIP3 for CCK
%%% interference in Wi-Fi standard
clear all
CCK_BB_spectrum        % load CCK baseband spectrum with 3 side
                       % lobes (44MHz wide)
CCK=[CCK(end:-1:1)' CCK']';          % shift CCK spectrum to 44MHz
CCK=CCK/max(abs(CCK)); % normalize CCK spectrum

%3rd order harmonic
HD3=conv(conv(CCK,CCK),CCK); % HD3 spectrum
figure(1)
hold off
freq=linspace(0,3*4*22e6,length(HD3));      % frequency span of HD3
semilogy(freq,HD3)       % HD3 spectrum plot
HD3_center=length(HD3)/2;
CCK_lobe=length(CCK)/8;   % width of CCK side lobe

%                        finding the edges of the band of the HD3 spectrum that
passes
%                        through the LPF
left_edge=fix(HD3_center-(25+6)/11*CCK_lobe);
right_edge=fix(HD3_center-(25-6)/11*CCK_lobe);

Wn1 = 2*pi*6e6;          % LPF bandwidth
[B1 A1] = butter(5,Wn1,'s');
% filter centered at 25MHz from the HD3 center
W1=linspace(2*pi*(25-3*4*11)*1e6,2*pi*(25+3*4*11)*1e6,length(HD3));
H1=abs(freqs(B1, A1, W1));  % filter frequency response

HD3_inband=H1'.*HD3;% filtered HD3
hold on
semilogy(freq,HD3_inband,'r')          % plot filtered HD3
Pin=sum(CCK.^2)*4*22e6/length(CCK);     % calculate CCK power
%                        generate two tones with the same power as CCK
%                        such that HD3 falls inband
twotone=zeros(1,length(CCK));
```

```
%%% This code is to calculate the equivalent IIP2 for CCK
%%% interference in Wi-Fi standard
clear all
CCK_BB_spectrum        % load CCK baseband spectrum with 3 side lobes
                       % 44MHz wide)
CCK=[CCK(end:-1:1)' CCK']';          % shift CCK spectrum to 44MHz
CCK=CCK/max(abs(CCK)); % normalize CCK spectrum

%3rd order harmonic
HD3=conv(conv(CCK,CCK),CCK); % HD3 spectrum
figure(1)
hold off
freq=linspace(0,3*4*22e6,length(HD3));      % frequency span of HD3
semilogy(freq,HD3)       % HD3 spectrum plot
HD3_center=length(HD3)/2;
CCK_lobe=length(CCK)/8;   % width of CCK side lobe

%                        finding the edges of the band of the HD3 spectrum that
passes
%                        through the LPF
left_edge=fix(HD3_center-(25+6)/11*CCK_lobe);
right_edge=fix(HD3_center-(25-6)/11*CCK_lobe);

Wn1 = 2*pi*6e6;          % LPF bandwidth
[B1 A1] = butter(5,Wn1,'s');
% filter centered at 25MHz from the HD3 center
W1=linspace(2*pi*(25-3*4*11)*1e6,2*pi*(25+3*4*11)*1e6,length(HD3));
H1=abs(freqs(B1, A1, W1));  % filter frequency response

HD3_inband=H1'.*HD3;% filtered HD3

hold on

semilogy(freq,HD3_inband,'r')          % plot filtered HD3
Pin=sum(CCK.^2)*4*22e6/length(CCK); % calculate CCK power
%                        generate two tones with the same power as CCK
%                        such that HD3 falls inband
twotone=zeros(1,length(CCK));
twotone(fix((44-1*25/3)/11*CCK_lobe))=sqrt(Pin);
```

```
%%% This code is to calculate the equivalent LO phase noise for CCK
%%% interference in Wi-Fi standard
clear all
CCK_BB_spectrum       % load CCK baseband spectrum with 3 side lobes (44MHz
                      % wide)
CCK=[CCK(end:-1:1)' CCK']';                    % shift CCK spectrum to 44MHz
CCK=CCK/max(abs(CCK));                         % normalize CCK spectrum
CCK_center=length(CCK)/2;
CCK_lobe=length(CCK)/8;                        % width of CCK side lobe
% shift CCK spectrum to 25MHz
CCK=CCK(CCK_center-25/11*CCK_lobe:CCK_center+25/11*CCK_lobe);
semilogy(CCK)
% Generate a single tone at 25MHz
single_tone=zeros(size(CCK));
single_tone(end/2)=sqrt(sum(CCK.^2)*2*25e6/length(CCK));
single_tone=single_tone+1e-10;      % add eps to the tone to plot it in log scale
figure(1)
hold off
freq=linspace(0,2*25e6,length(CCK));  % Frequency span of CCK spectrum=25MHz
PN25MHz=-99.2124;                     % PN at 25MHz in dBc/Hz
K=25e6*10^(PN25MHz/10);               %PN constant
PN=sqrt(K./freq);                     % Phase noise spectrum
PN(1)=1*PN(2);                        % avoid dividing by zero
semilogy(freq,CCK)                    % plot CCK spectrum
hold on
semilogy(freq,PN) % plot PN
semilogy(freq,single_tone,'-')        % plot the single tone

No_CCK=conv(PN(end:-1:1), CCK);    % Noise due to the CCK interferer
No_single_tone=conv(PN(end:-1:1), single_tone);     % noise due to the single tone
figure(2)
freq2=linspace(-2*25e6,2*25e6,length(No_CCK));   % frequency span of the noise
hold off
semilogy(freq2,No_CCK)                % plot noise due to CCK interferer
hold on
semilogy(freq2,No_single_tone)        % plot noise due to single tone interferer
% find the noise band edges passed through the LPF
N=length(No_CCK);
left_edge=N/2-6/100*N;
right_edge=N/2+6/100*N;

% calculate equivalent PN for the CCK interferer
PNeff = 10*log10(sum(No_CCK(left_edge:right_edge).^2)/sum(CCK.^2))-
10*log10(2*6e6)
```

APPENDIX D

MATLAB CODES FOR SYSTEM LEVEL CALCULATIONS

D.1. Bluetooth System Design MATLAB Code

```
%This file is for Bluetooth receiver system level design%
%Author: Ahmed Emira%
%Below is the flow chart of the system level design
%    _____________________        _____________________
%    |      Baseband      |       |     Standard      |
%    |    Requirements    |       |    Specifications |
%    |  (SNRmin, C/Imin)  |       |___________________|
%    |____________________|                /
%                \                         /
%                  \                      /
%                    \                   /
%                      \                /
%                       _____________________
%                      |     Derive required |
%                      |     system level specs |
%                      |_____________________|
%                                 |
%                      ___________\|/_________
%                      |  Blocks' specifications  | _______
%                      |__________________________|        |
%                                 |                        |
%                      ___________\|/_________             |
%                      | Calculate system specs |          /|\
%                      |________________________|           |
%                                 |                        |
%                    _____________\|/_____________          | NO
%                   | Do we meet required specs  |_____ |
%                   |_____________________________| |
%                                 | YES
%                          _______\|/_____
%                         |     Stop ☺     |
%                         |________________|
```

```
% All IIP3's are in dBm referenced to 50Ohm, unless otherwise specified
% All NF's are in dB referenced to 50Ohm, unless otherwise specified
% All gains are in dB unless otherwise specified
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%% Constants %%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
K = 1.38e-23; % Boltzmann constant
T = 300; % room temperature
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%% Baseband Requirements %%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
SNR=13; % dB
C2I=13; % minimum Carrier to inband interference ratio (dB)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%% Standard specifications %%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Sensitivity = -84; % dBm, 15dB better than standard
max_signal = -20; % dBm
Rb = 1e6; %Mbps
BW=0.6e6; %MHz
Eb_No=SNR-10*log10(Rb/BW);
% Adjacentchannel specs
Signal_level_C2I=-70+3; %dBm
C2I_3MHz = -40; %dBm
Interferer_level_C2I= Signal_level_C2I-C2I_3MHz;
Signal_level_IM3=-70+6; %dBm
Interferer_level_IM3 = -39; %dBm
C2I_IM3 = Signal_level_IM3-Interferer_level_IM3;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%% General parameters %%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Signal_ADC_max = 4; % this is the maximum signal level at ADC input(dBm)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%% Required system specifications %%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
NF_margin = 6;
Req_syst_NF = Sensitivity - (-174+10*log10(Rb)) - Eb_No - NF_margin

IIP3_margin = 3;
IM3 = Signal_level_IM3-C2I;
Req_syst_IIP3 = (3*Interferer_level_IM3-IM3)/2+IIP3_margin
Req_syst_IIP2 = 2*Interferer_level_C2I-Signal_level_C2I+C2I-6
                % 6dB is subtracted since it is a single tone test
                % No IP2 test is specified

Rx_Gain_max = Signal_ADC_max-Sensitivity;
Rx_Gain_min = Signal_ADC_max-max_signal;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%  Block specifications  %%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% indices [RF_filter = 1, LNA = 2, Mixer = 3, Filter = 4, VGA = 5]
NB=6; %number of blocks
RF_filer = 1; % indices
LNA = 2;
Mixer = 3;
Filter = 4;
VGA = 5;
ADC = 6;
ADC_SNR = 60; % in dB
ADC_NF = Signal_ADC_max - ADC_SNR - 10*log10(4*K*T*BW/2*1000);

%             RF_filter  LNA   Mixer  Filter  VGA   ADC
IIP2    =     [100       10    47     63      30    40];     % in dBm
IIP3    =     [100       -8    5      23      10    10];     % in dBm
NF_Gmax = [0             3     20     36      30    ADC_NF]; % in dB at max gain
NF_Gmin =  [0            15    20     36      40    ADC_NF]; % in dB at min gain
Gain_max = [0            15    16     6       24    0  ];    % max gain in dB
Gain_min =  [0           -15   16     6       0     0  ];    % min gain in dB
Atten_3M = [0            0     0      60      0     0];      % interference
                                                              % attenuation
                                                              % relative to the
                                                              % max gain
Mixer_feedthrough = -30; %RF to IF isolation in dB
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%  LO phase noise  %%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
LO_PN = -Interferer_level_C2I-Eb_No -10*log10(Rb)
+10*log10(10^(Signal_level_C2I/10) - 10^(Sensitivity/10))-NF_margin
                                % Phase noise in dBc/Hz at 25MHz
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%  System specifications  %%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=0:NB
   Gain_max_accum(i+1) = sum(Gain_max(1:i));  % accumulated gain from antenna
end                                           % to block input

for i=0:NB-1
   Int_atten_accum(i+1) = sum(Atten_3M(1:i)); % accumulated gain from antenna
end                                           % to block input for the interferer
IIP3_eff = IIP3+1.5*Int_atten_accum; % effective IP3
IIP2_eff = IIP2+2*Int_atten_accum; % effective IP2
IIP2_eff(LNA) = IIP2_eff(LNA) - (Mixer_feedthrough - Gain_max(Mixer)); % IM2
                               % of the LNA is attenuated by the mixer's feedthrough
```

```
IIP2_IR_Lin = 10.^((IIP2_eff-Gain_max_accum(1:NB))/20);  % IIP2 referred to the
                                                         % Rx input in SQRT(mW)
IIP3_IR_Lin = 10.^((IIP3_eff-Gain_max_accum(1:NB))/10);  % IIP3 referred to the
                                                         % Rx input in mW
NF_IR_Lin = (10.^(NF_Gmax/10)-1)./10.^(Gain_max_accum(1:NB)/10)+1; % NF
                                                         % referred to
                                                         % the Rx input in
                                                         % linear units

Syst_NF = 10*log10(sum(NF_IR_Lin-1)+1)          % System NF in dB

Syst_IIP3 = 10*log10(1/sum(1./IIP3_IR_Lin))     % System IIP3 in dBm

Syst_IIP2 = 20*log10(1/sum(1./IIP2_IR_Lin))     % System IIP2 in dBm

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%% IIP2, IIP3, and Noise contributions of different blocks %%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
IIP2_contributions = 10^(Syst_IIP2/20)./IIP2_IR_Lin(2:end);
figure
LABELS_IIP2 = {'LNA','Mixer','Filter','VGA'};
pie(IIP2_contributions, LABELS_IIP2);
title('IIP2 contributions of different blocks')

NF_contributions = (NF_IR_Lin-1)/(10.^(Syst_NF/10) -1);
LABELS_NF = {'RF Filter','LNA','Mixer','Filter','VGA'};
figure
pie(NF_contributions, LABELS_NF);
title('Noise contributions of different blocks')

IIP3_contributions = 10.^(Syst_IIP3/10)./IIP3_IR_Lin(2:end);
figure
LABELS_IIP3 = {'LNA','Mixer','Filter','VGA'};
pie(IIP3_contributions, LABELS_IIP3);
title('IIP3 contributions of different blocks')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%  Test scenarios  %%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Sensitivity test %%%%
Sig_level_IR = Sensitivity; % signal level at antenna (Rx input referred)
Noise_level_IR = -174+10*log10(BW); % thermal noise at antenna
Sig_level(1:NB) =  Sig_level_IR+Gain_max_accum(1:NB); % signal level at each
                                                      % block input
for i=0:NB-1
   NF_IR_accum_Lin(i+1)=sum(NF_IR_Lin(1:i)-1)+1;
end
```

```
Noise_level(1:NB) =
10*log10(10.^((Noise_level_IR+Gain_max_accum(1:NB))/10).*NF_IR_accum_Lin);
SNR = Sig_level-Noise_level;

figure
plot([0:NB-2],Sig_level(2:end), 'k-o',[0:NB-2], Noise_level(2:end), 'k-s')
set(gca,'XTick',[0:NB])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})
grid on
title('Sensitivity test')
ylabel('Input signal and noise levels in dBm')
legend('Signal level','Noise level',0)

figure
plot([0:NB-2], SNR(2:end), 'k-d')
set(gca,'XTick',[0:NB])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})
grid on
title('Sensitivity test')
ylabel('Input SNR level in dB')
legend('SNR',0)

%%%% Adjacent channel test (signal is 6dB above sensitivity) %%%%
Sig_level_IR = Signal_level_C2I; % desired signal level
Int_level_IR = Interferer_level_C2I; % interference level at 3MHz offset


for i=0:NB-1
   Gain_accum(i+1) = sum(Gain_max(1:i)); % accumulated voltage gain from
                                         % antenna to block input
end
Int_gain=Gain_max - Atten_3M; % maximum gain for the interferer
for i=0:NB-1
   Int_gain_accum(i+1) = sum(Int_gain(1:i)); % accumulated voltage gain from
end                                          % antenna to block input
Sig_level(1:NB) =  Sig_level_IR+Gain_accum(1:NB);
Int_level(1:NB) =  Int_level_IR+Int_gain_accum(1:NB);

figure
plot([1:NB-1],Sig_level(2:end), 'k-o',[1:NB-1], Int_level(2:end), 'k-v')
set(gca,'XTick',[1:NB-1])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})
grid on
title('Adjacent channel test in Bluetooth mode')
ylabel('Signal level in dBm')
legend('Signal level','Interferer level',0)
```

```
%%% maximum signal test %%%
Sig_level_IR = max_signal;
for i=0:NB
   Gain_accum_min(i+1) = sum(Gain_min(1:i)); % accumulated voltage gain from
end                                          % antenna to block input
Sig_level(1:NB) =  Sig_level_IR+Gain_accum_min(1:NB);

figure
plot([0:NB-2],Sig_level(2:end), 'k-s')
set(gca,'XTick',[0:NB-1])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})
grid on
title('Maximum signal test in Bluetooth mode')
ylabel('Signal level in dBm')
legend('Signal level',0)

%%% Signal at Sensitivity+51dB (LNA in Hi gain and VGA in Lo gain)%%%
Sig_offset = 51;
Sig_level_IR = Sensitivity+Sig_offset;
Gain=[Gain_max(1:NB-2) Gain_min(NB-1:NB)];
NF=[NF_Gmax(1:NB-2) NF_Gmin(NB-1:NB)];
for i=0:NB
   Gain_accum(i+1) = sum(Gain(1:i)); % accumulated voltage gain from antenna
End                                  % to block input
Sig_level(1:NB) =  Sig_level_IR+Gain_accum(1:NB);
NF_IR_Lin = (10.^(NF/10)-1)./10.^(Gain_accum(1:NB)/10)+1; % NF referred to the
                                                          % Rx input in linear units

Noise_level_IR = -174+10*log10(BW); % thermal noise at antenna
for i=0:NB-1
   NF_IR_accum_Lin(i+1)=sum(NF_IR_Lin(1:i)-1)+1;
end
Noise_level(1:NB) =
10*log10(10.^((Noise_level_IR+Gain_accum(1:NB))/10).*NF_IR_accum_Lin);

figure
plot([1:NB-1],Sig_level(2:end), 'k-o',[1:NB-1], Noise_level(2:end), 'k-s')
set(gca,'XTick',[1:NB])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})
grid on
title(' Gain switching test when LNA gain is Hi and VGA gain is Lo')
ylabel('Input signal and noise levels in dBm')
legend('Signal level','Noise level',0)
```

```
% Signal at Sensitivity+51dB (LNA in Lo gain and VGA in Lo gain)
Sig_offset = 51;
Sig_level_IR = Sensitivity+Sig_offset;
Gain=Gain_min;
NF=NF_Gmin;
for i=0:NB
   Gain_accum(i+1) = sum(Gain(1:i)); % accumulated voltage gain from antenna
End                                  % to block input

Sig_level(1:NB) =  Sig_level_IR+Gain_accum(1:NB);
NF_IR_Lin = (10.^(NF/10)-1)./10.^(Gain_accum(1:NB)/10)+1; % NF referred to the
                                                          % Rx input in
Noise_level_IR = -174+10*log10(BW); % thermal noise at antenna
for i=0:NB-1
   NF_IR_accum_Lin(i+1)=sum(NF_IR_Lin(1:i)-1)+1;
end
Noise_level(1:NB) =
10*log10(10.^((Noise_level_IR+Gain_accum(1:NB))/10).*NF_IR_accum_Lin);

figure
plot([1:NB-1],Sig_level(2:end), 'k-o',[1:NB-1], Noise_level(2:end), 'k-s')
set(gca,'XTick',[1:NB])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})
grid on
title(' Gain switching test when LNA gain is Lo and VGA gain is Lo')
ylabel('Input signal and noise levels in dBm')
legend('Signal level','Noise level',0)
```

D.2. Wi-Fi System Design MATLAB Code

```
%This file is for WiFi receiver system level design%
%Author: Ahmed Emira%
%Below is the flow chart of the system level design
%
%     _________________          _____________________
%     |    Baseband   |          |     Standard      |
%     |  Requirements |          |   Specifications  |
%     | (SNRmin, C/Imin) |       |___________________|
%     |_______________|                  /
%                  \                    /
%                    \                 /
%                      \              /
%                        \          /
%                  _____________________
%                  |  Derive required  |
%                  |  system level specs |
%                  |___________________|
%                            |
%                            |
%                  __________\|/_________
%                  | Blocks' specifications | _______
%                  |________________________|        |
%                            |
%                            |                       |
%                  __________\|/_________            |
%                  | Calculate system specs |       /|\
%                  |________________________|        |
%                            |                       |
%                            |                       |
%                 ___________\|/____________         | NO
%                | Do we meet required specs |_____ |
%                |___________________________|
%                            |
%                            | YES
%                         _____\|/_____
%                        |    Stop ☺    |
%                        |______________|
%
% All IIP3's are in dBm referenced to 50Ohm, unless otherwise specified
% All NF's are in dB referenced to 50Ohm, unless otherwise specified
% All gains are in dB unless otherwise specified
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%% Constants %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
K = 1.38e-23; % Boltzmann constan
T = 300; % room temperature

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%% Baseband Requirements %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Eb_No=13; % dB
C2I=13; % minimum Carrier to inband interference ratio (dB)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%% Standard specifications %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Sensitivity = -80; % dBm
max_signal = -4; % dBm
Rb = 11e6; %Mbps
BW=6e6; %MHz

% Adjacent channel specs
Signal_level_C2I=-70; %dBm
C2I_25MHz = -35; % dB
Interferer_level_C2I = Signal_level_C2I-C2I_25MHz;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%% General parameters %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Signal_ADC_max = 10; % this is the maximum signal level at ADC input(dBm)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%% Required system specifications %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
NF_margin = 6;
Req_syst_NF = Sensitivity - (-174+10*log10(Rb)) - Eb_No - NF_margin

IIP3_margin = 3;
IM3 = Signal_level_C2I-C2I;

Req_syst_IIP3 = (3*Interferer_level_C2I-IM3-37)/2+IIP3_margin
                        % 37dB is subtracted since the interference
                        % is a wideband CCK signal, not two tones.
                        % No IP3 test is specified
```

```
Req_syst_IIP2 = 2*Interferer_level_C2I-Signal_level_C2I+C2I-20
                    % 20dB is subtracted since the interference
                    % is a wideband CCK signal, not two tones.
                    % No IP2 test is specified

Req_syst_IIP2 = 2*Interferer_level_C2I-Signal_level_C2I+C2I-20
                    % 20dB is subtracted since the interference
                    % is a wideband CCK signal, not two tones.
Rx_Gain_max = Signal_ADC_max-Sensitivity;
Rx_Gain_min = Signal_ADC_max-max_signal;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%  Block specifications  %%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%indices [RF_filter = 1, LNA = 2, Mixer = 3, Filter = 4, VGA = 5]

NB=6; %number of blocks
RF_filer = 1; %indeces
LNA = 2;
Mixer = 3;
Filter = 4;
VGA = 5;
ADC = 6;
ADC_SNR = 42; % dB, assumed for now!
ADC_NF = Signal_ADC_max - ADC_SNR - 10*log10(4*K*T*BW/2*1000);

%           RF_filter   LNA   Mixer   Filter  VGA  ADC
IIP2   =    [100        11    48      64      31   41]      % in dBm
IIP3   =    [100        -8    5       23      10   10]      % in dBm
NF_Gmax =   [0          3     15      32      30   ADC_NF]  % at max gain,in dB
NF_Gmin =   [0          15    15      32      40   ADC_NF]  % at min gain in dB
Gain_max =  [0          15    16      6       43   0 ]      % max gain in dB
Gain_min =  [0          -15   16      6       0    0 ]      % min gain in dB
Atten_25M = [0          0     0       50      0    0 ]      % interference
                                                                % attenuation
                                                                % relative to the
                                                                % max gain
ADC_ENOB = 6; % assumed for now!
Mixer_feedthrough = -30; %RF to IF isolation in dB
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%  LO phase noise  %%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
LO_PN = -Interferer_level_C2I-Eb_No-
10*log10(Rb)+10*log10(10^(Signal_level_C2I/10)-10^(Sensitivity/10))-NF_margin
                                % Phase noise in dBc/Hz at 25MHz
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%  System specifications  %%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=0:NB
   Gain_max_accum(i+1) = sum(Gain_max(1:i)); % accumulated voltage gain from
antenna
                                  % to block input
end
for i=0:NB-1
   Int_atten_accum(i+1) = sum(Atten_25M(1:i));  % accumulated gain from antenna
end                                                 % to block input for the interferer

IIP3_eff = IIP3+3*Int_atten_accum;                 % effective IP3
IIP2_eff = IIP2+2*Int_atten_accum;                 % effective IP2

% IM2 of the LNA is attenuated by the mixer's feedthrough
IIP2_eff(LNA) = IIP2_eff(LNA) - (Mixer_feedthrough - Gain_max(Mixer));

% IIP2 is done when the signal is -70dBm so VGA gain is reduced by -70-(-80)=10dB
Gain_IIP2 = Gain_max_accum(1:NB)+[0 0 0 0 0 -10];

IIP2_IR_Lin = 10.^((IIP2_eff-Gain_IIP2)/20);    % IIP2 referred to the Rx input in
                                                % sqrt(mW)

IIP3_IR_Lin = 10.^((IIP3_eff-Gain_max_accum(1:NB))/10);  % IIP3 referred to the
                                                          % Rx input in sqrt(mW)

NF_IR_Lin = (10.^(NF_Gmax/10)-1)./10.^(Gain_max_accum(1:NB)/10)+1;
                                             % NF referred to Rx input in
                                             % linear units

Syst_NF = 10*log10(sum(NF_IR_Lin-1)+1)        % System NF in dB

Syst_IIP2 = 20*log10(1/sum(1./IIP2_IR_Lin))      % System IIP2 in dBm

Syst_IIP3 = 10*log10(1/sum(1./IIP3_IR_Lin)) % System IIP3 in dBm

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%  Test scenarios  %%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% Sensitivity test %%%
Sig_level_IR = Sensitivity; % signal level at antenna (Rx input referred)
Noise_level_IR = -174+10*log10(BW); % thermal noise at antenna
Sig_level(1:NB) =  Sig_level_IR+Gain_max_accum(1:NB);  % signal level at each
                                                        % block input
```

```
for i=0:NB-1
   NF_IR_accum_Lin(i+1)=sum(NF_IR_Lin(1:i)-1)+1;
end
%Noise_level(1) = Noise_level_IR;
Noise_level(1:NB) =
10*log10(10.^((Noise_level_IR+Gain_max_accum(1:NB))/10).*NF_IR_accum_Lin);

SNR = Sig_level-Noise_level;

figure
plot([0:NB-2],Sig_level(2:end), 'k-o',[0:NB-2], Noise_level(2:end), 'k-s')

set(gca,'XTick',[0:NB])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})

grid on
title('Sensitivity test')
ylabel('Input signal and noise levels in dBm')
legend('Signal level','Noise level',0)

figure
plot([0:NB-2], SNR(2:end), 'k-d')
set(gca,'XTick',[0:NB])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})
grid on
title('Sensitivity test')
ylabel('Input SNR level in dB')
legend('SNR',0)

%%% Adjacent channel test (signal is 6dB above sensitivity) %%%
Sig_level_IR = Signal_level_C2I;            % desired signal level
Int_level_IR = Interferer_level_C2I;        % interference level at 25MHz offset
Gain=Gain_max;
Gain(VGA)=Gain_max(VGA)-(Sig_level_IR-Sensitivity)

for i=0:NB-1
   Gain_accum(i+1) = sum(Gain(1:i)); % accumulated voltage gain from antenna
end                                  % to block input

Int_gain=Gain - Atten_25M;           % maximum gain for the interferer
for i=0:NB-1
   Int_gain_accum(i+1) = sum(Int_gain(1:i));      % accumulated voltage gain from
end                                               % antenna to block input

Sig_level(1:NB) =  Sig_level_IR+Gain_accum(1:NB);
Int_level(1:NB) =  Int_level_IR+Int_gain_accum(1:NB);
```

```
figure
plot([1:NB-1],Sig_level(2:end), 'k-o',[1:NB-1], Int_level(2:end), 'k-v')
set(gca,'XTick',[1:NB-1])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})
grid on
title('Adjacent channel test in WiFi mode')
ylabel('Signal level in dBm')
legend('Signal level','Interferer level',0)

%%% maximum signal test %%%
Sig_level_IR = max_signal;
for i=0:NB
   Gain_accum_min(i+1) = sum(Gain_min(1:i)); % accumulated voltage gain from
end                                          % antenna to block input

Sig_level(1:NB) =  Sig_level_IR+Gain_accum_min(1:NB);
figure
plot([0:NB-2],Sig_level(2:end), 'k-s')
set(gca,'XTick',[0:NB-1])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})
grid on
title('Maximum signal test in WiFi mode')
ylabel('Signal level in dBm')
legend('Signal level',0)

%%% Signal at Sensitivity+20dB (the LNA in Hi gain) %%%
Sig_offset = -37-Sensitivity;
Sig_level_IR = Sensitivity+Sig_offset;
Gain=Gain_max;
Gain(VGA)=Gain_max(VGA)-Sig_offset;
NF=NF_Gmax;
NF(VGA)=NF_Gmin(VGA);
for i=0:NB
   Gain_accum(i+1) = sum(Gain(1:i));    % accumulated voltage gain from antenna
End                                     % to block input

Sig_level(1:NB) =  Sig_level_IR+Gain_accum(1:NB);
NF_IR_Lin = (10.^(NF/10)-1)./10.^(Gain_accum(1:NB)/10)+1;  % NF referred to the
                                                           % Rx input in linear units

Noise_level_IR = -174+10*log10(BW);     % thermal noise at antenna

for i=0:NB-1
   NF_IR_accum_Lin(i+1)=sum(NF_IR_Lin(1:i)-1)+1;
end
```

```
Noise_level(1:NB) =
10*log10(10.^((Noise_level_IR+Gain_accum(1:NB))/10).*NF_IR_accum_Lin);

figure
plot([1:NB-1],Sig_level(2:end), 'k-o',[1:NB-1], Noise_level(2:end), 'k-s')
set(gca,'XTick',[1:NB])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})
grid on
title(' Gain switching test when LNA gain is Hi and VGA gain is Lo')
ylabel('Input signal and noise levels in dBm')
legend('Signal level','Noise level',0)

%%% Signal at Sensitivity+20dB (the LNA in Lo gain) %%%
Sig_offset = -37-Sensitivity;
Sig_level_IR = Sensitivity+Sig_offset;

Gain=Gain_min;
Gain(VGA)=Gain_max(VGA)-Sig_offset+Gain_max(LNA)-Gain_min(LNA);
NF=NF_Gmin;
for i=0:NB
   Gain_accum(i+1) = sum(Gain(1:i)); % accumulated voltage gain from antenna
end                                  % to block input
Sig_level(1:NB) =  Sig_level_IR+Gain_accum(1:NB);
NF_IR_Lin = (10.^(NF/10)-1)./10.^(Gain_accum(1:NB)/10)+1; % NF referred to the
                                                          % Rx input in linear
                                                          % units
Noise_level_IR = -174+10*log10(BW);     % thermal noise at antenna
for i=0:NB-1
   NF_IR_accum_Lin(i+1)=sum(NF_IR_Lin(1:i)-1)+1;
end
Noise_level(1:NB) =
10*log10(10.^((Noise_level_IR+Gain_accum(1:NB))/10).*NF_IR_accum_Lin);
figure
plot([1:NB-1],Sig_level(2:end), 'k-o',[1:NB-1], Noise_level(2:end), 'k-s')
set(gca,'XTick',[1:NB])
set(gca,'XTickLabel',{'LNA';'Mixer';'Filter';'VGA';'ADC'})
grid on
title(' Gain switching test when LNA gain is Lo and VGA gain is Lo')
ylabel('Input signal and noise levels in dBm')
legend('Signal level','Noise level',0)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%% IIP2, IIP3, and Noise contributions of different blocks %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
IIP2_contributions = 10^(Syst_IIP2/20)./IIP2_IR_Lin(2:end);
```

```
figure
LABELS_IIP2 = {'Mixer','Filter', 'LNA+VGA+ADC'};
pie([IIP2_contributions(Mixer-1) IIP2_contributions(Filter-1)
      IIP2_contributions(LNA-1)+IIP2_contributions(VGA-1)+
      IIP2_contributions(ADC-1)], LABELS_IIP2);
title('IIP2 contributions of different blocks')

IIP3_contributions = 10.^(Syst_IIP3/10)./IIP3_IR_Lin(2:end);
figure
LABELS_IIP3 = {'LNA','Mixer','Filter', 'VGA', 'ADC'};
pie(IIP3_contributions, LABELS_IIP3);
title('IIP3 contributions of different blocks')

NF_contributions = (NF_IR_Lin-1)/(10.^(Syst_NF/10) -1);
LABELS_NF = {'RF Filter','LNA','Mixer','Filter', 'VGA', 'SD ADC'};
figure
pie(NF_contributions, LABELS_NF);
title('Noise contributions of different blocks')

%%% Power consumption and area contributions in WiFi mode %%%
Power_contributions_WiFi = [29.2 26.9 5.8 4.5 33.5];
figure
LABELS_power = {'LNA/Mixer','PLL', 'Filter', 'VGA', 'ADC'};
pie(Power_contributions_WiFi, LABELS_power);
title('Power consumption contributions of different blocks in WiFi mode')

Area_contributions_WiFi = [4.5 13.2 15.9 8.1 58.3];
figure
LABELS_area = {'LNA/Mixer','PLL', 'Filter', 'VGA', 'ADC'};
pie(Area_contributions_WiFi, LABELS_area);
title('Area contributions of different blocks in WiFi mode')
```

APPENDIX E

LIST OF PAPERS USED IN THIS DISSERTATION

E.1. Journal Papers

1. Ahmed N. Mohielding, Ahmed Emira, Edgar Sánchez-Sinencio, "A 100MHz, 8mW ROM-Less Quadrature Direct Digital Frequency Synthesizer," *IEEE J. Solid-State Circuits*, vol. 37, no. 10, pp. 1235-1243, Oct. 2002.
2. Wenjun Sheng, Bo Xia, Ahmed Emira, Chunyu Xin, Ari Y. Valero-López, Sung T. Moon, and Edgar Sánchez-Sinencio, "A 3V, 0.35μm CMOS Bluetooth Receiver IC," *IEEE J. Solid-State Circuits*, vol. 38, no. 1, pp. 30-42, Jan. 2003.
3. Ahmed Emira and Edgar Sánchez-Sinencio, "A Low-Power CMOS Complex Filter with Frequency Tuning," *IEEE Trans. Circuits and Syst. II,* vol. 50, no. 10, pp. 742-754, Oct. 2003.
4. Ahmed Emira, *Alberto* Valdes-Garcia, Bo Xia, Ahmed Mohieldin, Ari Valero-Lopez, Sung Moon, Chunyu Xin, and Edgar Sánchez-Sinencio "Chameleon: A Dual-Mode 802.11b/Bluetooth Receiver in 0.25μm BiCMOS," to be submitted to *IEEE J. of Solid-State Circuits*.
5. Ahmed Emira and Edgar Sánchez-Sinencio, "Variable Gain Amplifier with Offset Cancellation," to be submitted to *IEEE J. of Solid-State Circuits*.

E.2. Conference Papers

1. Ahmed Emira, Alberto Valdes-Garcia, Bo Xia, Ahmed Mohieldin, Ari Valero-Lopez, Sung Moon, Chunyu Xin, and Edgar Sánchez-Sinencio . "A Dual-Mode 802.11b/Bluetooth Receiver in 0.25μm BiCMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers,* San Francisco, CA, Feb. 2004.
2. Ahmed Emira, Alberto Valdes-Garcia, and Edgar Sánchez-Sinencio, "System Level WLAN Receiver Design," accepted in *MidWest Symp. on Circuits and Syst.*, Cairo, Egypt, Dec. 2003.
3. Ahmed Emira and Edgar Sánchez-Sinencio, "A Low-Power Complex Filter for Bluetooth with Frequency Tuning," *IEEE Int. Symp. Circuits and Systems, ISCAS 2003*, Bangkok, Thailand, May 2003.
4. Ahmed Emira and Edgar Sánchez-Sinencio, "Variable Gain Amplifier with Offset Cancellation," *Great Lakes Symp. on VLSI, GLSVLSI 2003,* Washington DC, April 2003.
5. Wenjun Sheng, Bo Xia, Ahmed Emira, and Edgar Sánchez-Sinencio, "Integrated Receiver Design for Bluetooth", *IEEE CAS-Notre Dame Workshop on Wireless Communications and Networking,* University of Notre Dame, Aug. 2001.
6. Ahmed Emira and Edgar Sánchez-Sinencio, "Design Tradeoffs of CMOS Current Mirrors Using One-Equation for All-Region Model," *IEEE Int. Symp. Circuits and Systems, ISCAS 2002*, Scottsdale, Arizona, May 2002.

7. Ahmed N. Mohieldin, Ahmed Emira, Edgar Sánchez-Sinencio, "A 2v 11 Bit Incremental A/D Converter Using Floating Gate Technique," *IEEE Int. Symp. Circuits and Systems, ISCAS 2002*, Scottsdale, Arizona, May 2002.

8. Wenjun Sheng, Bo Xia, Ahmed Emira, Chunyu Xin, Ari Y. Valero-López, Sung T. Moon, and Edgar Sánchez-Sinencio, "A Fully-Integrated 0.35μm CMOS Bluetooth Receiver," *IEEE Custom Integrated Circuits Conference, CICC 2002,* Orlando, Florida, May 2002.

VITA

Ahmed A. Emira was born in Cairo, Egypt in 1974. He received the B.Eng. and M.Eng. degrees from the Electronics and Communications Department, Cairo University, Egypt, in 1997 and 1999, respectively. He worked as a teaching assistant in the same department from July 1997 until December 1999.

Mr. Emira worked towards his Ph.D. degree at the Analog and Mixed Signal Center, Department of Electrical Engineering, Texas A&M University, College Station, TX, since January 2000 and received his Ph.D. in December 2003. He held an internship at Motorola Inc., Austin, TX, from September 2001 till August 2002. His current research interests include mixed-signal circuits, analog filter design, and communications systems architectures.

Permanent address: 7001 South Congress Ave. #F202, Austin, TX 78745.