FREQUENCY SYNTHESIZER DESIGNS AND THEIR APPLICATIONS

Except where reference is made to the work of others, the work described in this thesis is my own or was done in collaboration with my advisory committee. This thesis does not include proprietary or classified information.

_____
Wenting Deng

Certificate of Approval:

_____	_____
Richard C. Jaeger Distinguished University Professor Electrical & Computer Engineering	Fa Foster Dai, Chair Associate Professor Electrical & Computer Engineering
_____	_____
Guofu Niu Professor Electrical & Computer Engineering	George T. Flowers Interim Dean Graduate School

FREQUENCY SYNTHESIZER DESIGNS AND THEIR APPLICATIONS

Wenting Deng

A Thesis

Submitted to

the Graduate Faculty of

Auburn University

in Partial Fulfillment of the

Requirements for the

Degree of

Master of Science

Auburn, Alabama
August 4, 2007

FREQUENCY SYNTHESIZER DESIGNS AND THEIR APPLICATIONS

Wenting Deng

_____

Signature of Author

_____

Date of Graduation

VITA

Wenting Deng was born in Nov 6th, 1981 in Chongqing, P. R. China. She attended Sichuan University in Sep 1999 and graduated with a Bachelor of Science degree in Microelectronics in July 2003. Then she joined China Electronics Technology Group Corporation, as a mixed signal circuit design engineer. In August 2005, she entered graduate program of the Electrical and Computer Engineering Department, Auburn University.

THESIS ABSTRACT

FREQUENCY SYNTHESIZER DESIGNS AND THEIR APPLICATIONS

Wenting Deng

M.S., Auburn University, August 4, 2007
(B.S., Sichuan University, 2003)

87 Typed Pages

Directed by Fa Foster Dai

This thesis discusses design and applications of two main frequency synthesizers: direct digital frequency synthesis (DDS) and phase lock loop (PLL) synthesis.

DDS is a quickly developed digital technique for frequency synthesis and waveform generation in modern communication systems. It provides many advantages including flexible phase and frequency adjustment, highly-stable and fast frequency conversion. A basic DDS system consists of three main building blocks: accumulator, sine look up table and a digital to analog converter (DAC). Actually the ROM for sine look-up table occupies the majority of the DDS area and limits its maximum operation frequency due to the delay through the multi-layer decoders. To reduce the power dissipation, a nonlinear DAC has been introduced to replace the ROM-look-up table and the linear DAC.

In most modern wireless communication systems, digital modulations become more and more popular since it increases channel capacity, the ability to transmit and receives information with higher accuracy than an analog communication system in the presence of noise and distortion. One of the most important applications of DDS is used in the digital modulations. It is very easy and flexible to implement phase modulations and frequency modulations by changing the accumulator's output or frequency control words respectively.

PLL also has been played an important role in communication systems. Compare to DDS with the same operation speed, PLL consumes smaller area and less power but has a relatively narrow tuning range due to voltage control oscillator's (VCO) "free running" [1]. And among kinds of PLLs, fractional-N PLL stands out clearly from others. It can have a wide loop bandwidth, fast settling time and small channel space. However the fractional spur is the drawback, it is even worse than in other PLL. In order to reduce the spurs, delta-sigma modulators was used to decrease the periodic disturbance without affecting the synthesizer's function. It moves noise from in-band to out-band, which can be removed by following low-pass filter.

A 5 GHz differential integer-N PLL is designed and fabricated in 0.5um Silicon Germanium technology. Circuit designs of main building blocks are discussed in details, such as phase detector, charge pump, 8 bit multi-modular divider with extension and programmable divider. And all digital blocks are used current-mode-logic (CML) for its high speed and relatively low power consumption.

ACKNOWLEDGMENTS

It has been a great privilege to be a graduate student in the Electrical and Computer Engineering department at Auburn University and work closely with my advisor Dr. Fa Foster Dai. He guided and encouraged me throughout my studies. And his keen insight into integrated circuit design is the key factor in the success of this research.

I would also like to thank my advisory committee members, Dr. Richard C. Jaeger and Dr. Guofu Niu for their guidance and advices on this work.

I greatly appreciate the help from my colleagues in Dr Dai's group who have made contributions to my research. I am especially indebted to Dayu Yang, Vasanth Kakani, Xuefeng Yu, Yuan Yao, Xueyang Geng, Desheng Ma, Yuehai Jin and Zhenqi Chen for their support and friendship throughout the course of this research.

Last but not the least, I am deeply grateful to my parents Yucheng Deng and Lizhong Yang for their love and support in my whole life. Their confidence in me and their high expectation of me are the motive force for me to go forward.

Style manual or journal used: IEEE Journal on Solid State Circuits

Computer software used: Microsoft Word 2003

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

1.1 Background Statement

Frequency synthesis is now so natural that every radio design uses only synthesized signals. It has entered a new age because of the huge acceleration of cellular telephony and the newly emerging markets of wireless and personal communications services. Many efforts are underway to increase the integration level of such circuits in low-cost technology. As a result the chip expenses have been dropped by technology improvement and better circuit design.

Meanwhile, on the other hand, frequency synthesis is a challenge area in IC design, as it includes both analog and digital technologies. To design a synthesizer one has to apply a great variety of disciplines such as oscillators, accumulators, digital-to-analog converters, amplifiers, filters, phase detectors and etc. A high performance, low-cost, low-power, small-area, long-battery-life solution for designs has been the dream for decades.

In fact, there are major advances in two key syntheses. The first one is Phase Locked Loop (PLL), which is the most popular and covers more than 95 percent of the frequency synthesis. In the past many years, the PLL frequency synthesizer can only generate exact multiple times of a reference. As a result, it limits the output accuracy of voltage-control oscillator and the reference frequency must be equal to the channel spacing [1]. In recent years, fractional-N type PLL has been used to solve the narrow loop bandwidth and low

frequency switching speed problems. Its output frequency can vary by a fraction of the input reference frequency, thus It allows the reference frequency to be much bigger than the channel spacing [2]. However, it also has its weakness -- fractional spurs in the output spectrum caused by periodic switching of the divider mode.

Although PLL is a conventional way to realize low phase noise, low power consumption frequency synthesizer, the need of high-Q [3] passive component and off chip components makes it hard for a full integration. Direct digital synthesis (DDS), the second main frequency synthesizer has been greatly developed recent years. It generates the signal digitally and converts it via a digital-to-analog converter (DAC) to a sine-kind wave. This synthesizer can provide fast frequency switching, fine resolution and wide tuning range in frequency-agile communication systems and it is suitable for integration for no need of off-chip components. Due to its own architecture, DDS can easily make various digital modulations such as BPSK, QPSK, MSK, and GMSK, which have wide range applications in wireless communications.

1.2 **Technology Used for Frequency Synthesizer**

With the rapid development of microelectronic technology, the transistor size enters deep submicron level. It largely increases the chip integration and operation speed of system.

For many years CMOS technology is a typical option for the low-gigahertz frequency design [4] [5]. It takes advantages in low power dissipation, low cost and easy scaling. However, its low speed, large noise, and the limited set of available passive devices really cumber CMOS in RFIC design. Bipolar complementary metal-oxide-semiconductor (BiCMOS) technology is then introduced. It can provide high

operation speed, high integration capability of both digital and analog circuit, high current-drive capability and high performance levels. In BiCMOS technology, the hetero-junction bipolar transistor uses two different materials such as Si and Ge to form a PN junction inside the transistor. It makes this technology reach higher unity gain bandwidth product (f_T), low noise, good linearity and low power consumption compared to BJT.

In this research work, we used 0.5 μm SiGe technology [6] [7] which is a mature BiCMOS technology providing a full device suite for RF IC design such as hetero-junction bipolar transistor (HBT), n-MOS & p-MOS, diffused and poly-silicon resistors, varactor diodes, spiral inductors and etc.

1.3 **Thesis Organization**

In Chapter 2, the concepts and components of direct digital synthesis including accumulator, sin-look-up table, and digital-analog converter are reviewed. An introduction of nonlinear DAC is also presented which solves the speed bottleneck caused by ROM lookup table and linear DAC.

In Chapter 3, the basics of digital modulations are introduced. And the main objective of this chapter is to discuss how to implement various phase and frequency modulations in DDS, plus how to filter the signal so as to reduce the transmitted bandwidth. Then a comparison and test result among these modulations will be given.

In Chapter 4, introduce basic concepts of PLL and generally discuss several types of PLL (Charge Pump PLL, Integer-N PLL and Fractional-N PLL) with their structures, operation theories advantages and shortages.

In Chapter 5, the RF circuit designs of PLL building blocks are presented: Phase/Frequency detector, Charge pump, programmable current setting, 3 bit Programmable divider, Multiple Modular Divider with extension. Furthermore CML digital circuits are introduced to replace trail to trail logic circuits in most digital blocks. Low power consumption, easy updating, smaller layout area, and higher operation speed are also mentioned.

In Chapter 6, Analysis and measurement results for fabricated PLL chip is presented (including PCB board design, test devices use). At last, conclude the thesis with a summary.

CHAPTER 2 DIRECT DIGITAL SYTHESIS OPERATION THERORY

2.1 Basic Concepts

In the last few years, direct digital synthesis (DDS) technology has captured the attention of synthesizer designers and enjoys an unusual popularity. Due to its notable virtues such as highly-stable, flexible phase and frequency adjustment, fast frequency conversion and low phase noise, DDS has been broadly used in communication system, electronic apparatus and instrument. Various designs and architectures are used to implement DDS. In fact, a standard DDS is a digital-and-analog mixed signal device and generates a sine wave at its output, not "all digital." shown as Figure 2-1 [1].

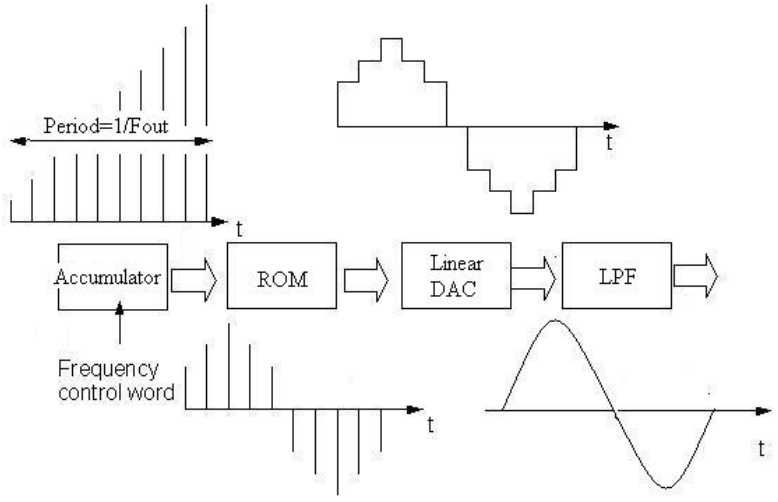

Figure 2-1 Conventional ROM-based DDS

The input to the accumulator, the first DDS element, is called frequency control word (FCW) which determines the output frequency of DDS. At each positive edge of the reference clock cycle the FCW is added to an accumulator. Since the bits of accumulator is constant and the value in the accumulator represents the phase of the

output sinusoid, if the FCW is large, the accumulator will be “filled” faster, then the period of sine wave will be shorter. Otherwise, if the FCW is small, the period of sine wave will extend. Therefore, FCW can be used for frequency control and modulation.

The accumulator, which is a discrete digital integrator, is used as the DDS indexer, controlled by the input word and the clock, generates a digital ramp. The output of the accumulator represents the sine wave’s phase. And the slope of the accumulator output can be viewed as the rate of phase change. As the phase increment is given by

$$\Delta\varphi = \frac{FCW \times 2\pi}{2^N} \qquad (2\text{-}1)$$

where N is the bit number of accumulator

Since at its output we have the digital signal that represents phase, it is rather simple to add phase control, just by putting an adder (or subtracter) between the accumulator and the ROM. Then phase modulation can be easily implemented.

The sine lookup table ROM converts the phase information to its amplitude presentation, performs the transformation in digital domain ($\varphi => \sin\varphi$). It stores the corresponding values of the sinusoid wave. This is also the block that limits the operation speed and large the chip size.

The DAC and filter following the ROM are used to transforms all the digital information to an analog signal. And the frequency resolution is determined by the bit number of the phase accumulator.

$$f_{out} = f_{clk} \cdot \frac{FCW}{2^N} \qquad (2\text{-}2)$$

where N is the bit number of phase accumulator

2.2 **Phase Accumulator**

The accumulator is one of the major elements of direct digital synthesizers. It adds up a frequency control word FCW at each clock cycle and generates a phase word of a sine wave at the output. Therefore, state 0 means phase 0 and state 2^{N-1} means 2π. And the bit number of accumulator determines the output resolution and frequency.

The accumulator is a digital integrator, performing the arithmetic function

$$P(n) = P(n-1) + L \tag{2-3}$$

where P(n) is a N bit word, (n) represents the n^{th} clock cycle, and L is the frequency control. The accumulator is usually constructed by adders and registers. The register is a storage device which changes its output only when clocked.

The N bit accumulator determines the output resolution frequency, given by

$$F_{out} = \frac{F_{ck}}{2^N} \tag{2-4}$$

Since all arithmetic operations are done modulo 2^N, any input L having a value $0 \le L \le 2^{N-1}$ has an equivalent input given by $L^* = 2^N - L$ that will yield exactly the same DDS output frequency. The two control inputs, *L* and *L**, are completely symmetric, and can be viewed as the same accumulation rate, one being clockwise and the other counterclockwise.

When it comes to accumulator's structure, it is usually constructed from similar blocks. The only connection between blocks is through the carrier bits. For example, a 24 bits accumulator can use six 4-bit similar blocks or twelve 2-bit adders, and other combinations. There are kinds of ways to implement accumulator. However, in order to

implement frequency modulations, the accumulator used in DDS need to use carry look-ahead structure.

As we know from binary addition, the carry in for bit 2 of the adder is exactly the carry out of bit 1, so the formula is

$$c_2 = (b_1 \times c_1) + (a_1 \times c_1) + (a_1 \times b_1) \tag{2-5}$$

Similarly, carry in for bit 1 is defined as

$$c_1 = (b_0 \times c_0) + (a_0 \times c_0) + (a_0 \times b_0) \tag{2-6}$$

Substituting the definition of c_1 for the first equation results in this formula:

$$\begin{aligned} c_2 = {} & (a_1 \times a_0 \times b_0) + (a_1 \times a_0 \times c_0) + (a_1 \times b_0 \times c_0) + (b_1 \times a_0 \times b_0) + (b_1 \times a_0 \times c_0) \\ & + (b_1 \times b_0 \times c_0) + (a_1 \times b_1) \end{aligned} \tag{2-7}$$

We can imagine how the equation expands as we get to higher bits in the adder; it grows exponentially with the number of bits. This complexity is reflected in the cost of hardware for fast carry, making this simple scheme prohibitively expensive for wide adders. Fortunately, carry-look-ahead adder limits the complexity of the equations to simplify the hardware, while still making substantial speed improvements over ripple carry. It relies on levels of abstraction in its implementation.

If we were to rewrite the equation for c_2 using this equation, we would see some repeated patterns:

$$c_2 = (a_1 \times b_1) + (a_1 + b_1) \times ((a_0 \times b_0) + (a_0 + b_0) \times c_0) \tag{2-8}$$

Note the repeated appearance of $(a_i \times b_i)$ and $(a_i + b_i)$ in the formula above. We let

$$g_i = a_i \times b_i \tag{2-9}$$

$$p_i = a_i + b_i \tag{2-10}$$

We can get $c_{i+1} = g_i + p_i \times c_i$, suppose g_i is 1, then c_{i+1}=1. That is, the adder generates a carry out (c_{i+1}) independent of the value of carry in (c_i). Now suppose that g_i is 0, and p_i is 1, then $c_{i+1} = c_i$. That is, the adder propagates carry in to a carry out.

$$c_1 = g_0 + (p_0 \times c_0) \tag{2-11}$$

$$c_2 = g_1 + (p_1 \times g_0) + (p_1 \times p_0 \times c_0) \tag{2-12}$$

$$c_3 = g_2 + (p_2 \times g_1) + (p_2 \times p_1 \times g_0) + (p_2 \times p_1 \times p_0 \times c_0) \tag{2-13}$$

These equations just represent common sense: c_i is a 1 if some earlier adder generates a carry and all intermediary adders propagate a carry. Therefore, the total delay of an N-bit accumulator is equal to $(N-1)t_{carry}+t_{sum}$, where t_{carry} is the time for carry generation and t_{sum} is the time for sum generation in last bit adder.

To go faster, we'll need carry-look-ahead at a higher level. Here, they are for three 4-bit adder blocks:

$$P_0 = p_3 \times p_2 \times p_1 \times p_0 \tag{2-14}$$

$$P_1 = p_7 \times p_6 \times p_5 \times p_4 \tag{2-15}$$

$$P_2 = p_{11} \times p_{10} \times p_9 \times p_8 \tag{2-16}$$

That is, (P_i) is true only if each of the bits in the group will propagate a carry.

$$G_0 = g_3 + (p_3 \times g_2) + (p_3 \times p_2 \times g_1) + (p_3 \times p_2 \times p_1 \times g_0) \tag{2-17}$$

$$G_1 = g_7 + (p_7 \times g_6) + (p_7 \times p_6 \times g_5) + (p_7 \times p_6 \times p_5 \times g_4) \tag{2-18}$$

$$G_2 = g_{11} + (p_{11} \times g_{10}) + (p_{11} \times p_{10} \times g_9) + (p_{11} \times p_{10} \times p_9 \times g_8) \tag{2-19}$$

For generate signal (G_i), we care only if there is a carry out of the most significant bit of the 4-bit group. This obviously occurs if generate is true for that most significant

bit; it also occurs if an earlier generate is true and all the intermediate propagates, including that of the most significant bit, are also true.

Figure 2-2 shows the architecture of carry look-ahead accumulator. It offers a faster path than waiting for the carries to ripple through all 1-bit adders. This faster path is paved by two signals, generate and propagate. The former creates a carry regardless of the carry input, and the other passes a carry along [8].

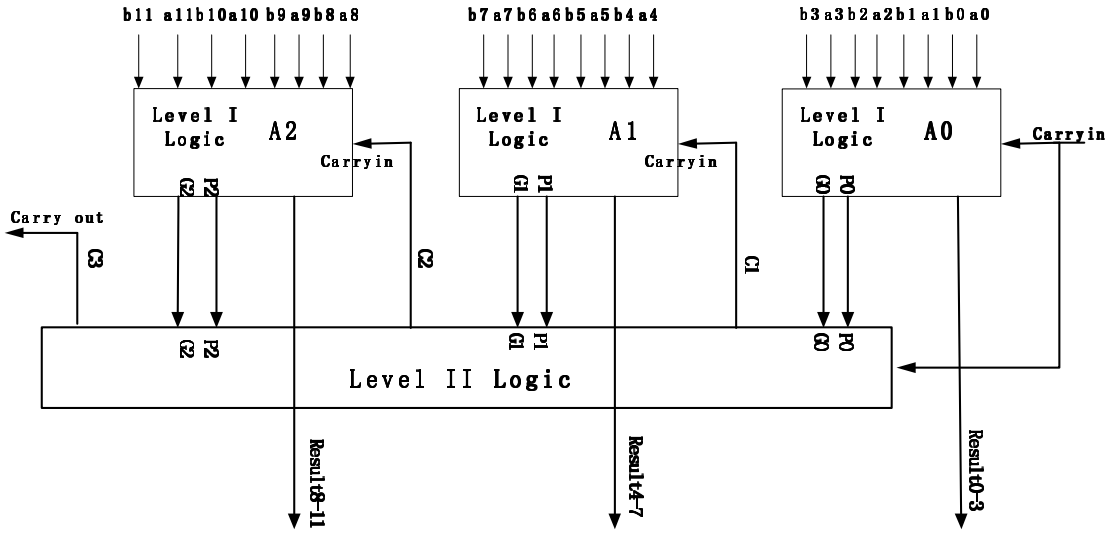

Figure 2-2 Carry-look-ahead accumulator structure

2.3 **Look Up Table and Sine Rom Compression**

2.3.1 Look Up Table ROM

In a direct digital synthesizer, the Rom is used as a lookup table to convert its phase input digital data bits to output digital amplitude data bits, to drive the DAC. The output of the accumulator is used as the address input of the lookup table and represents the sine-wave phase. This phase information needs to be converted to its amplitude value to drive the DAC and achieve the required analog output.

In typical RF applications, the ROM output must have a resolution in the range of 10 to 12 bits. It was suggested that increasing accumulator length N yields arbitrarily small steps in the output frequency—an important property of DDS. While the accumulator can be chosen to be relatively wide, the ROM may become prohibitively large and the power consumption is huge. For instance, if the widths of the accumulator and the ROM output words are 12 bits and 10 bits, respectively, then the ROM requires $2^{12\times10} = 4.1\times10^4$ cells. For this reason, the accumulator can still be designed with a wide output word so as to provide fine frequency steps, but only the most significant bits of this word are applied as ROM input.

However, if the ROM phase steps are not as small as those in the accumulator, a "phase truncation error" [9] [10] corrupts the output sinusoid. This type of error is also periodic, resulting in spurs. Assume N is the accumulator's size, W is the number of bits addressing the ROM. B=N-W is the truncated phase word bits. The number of spurs is:

$$M = \frac{2^{B-1}}{(FCW, 2^N)} \tag{2.20}$$

where (FCW, 2^N) represents the greatest common divisor of the FCW and 2^N.

2.3.2 Quadrant Compression of ROM

The size of the ROM can be reduced by more than 75 percent by taking advantage of the fact that only one quadrant of the sine wave need to be stored. For $0\leq\alpha\leq90$, $\sin(90-\alpha)=\sin(90+\alpha)$, $\sin(270-\alpha)=\sin(270+\alpha)$, $\sin(\alpha)= -\sin(-\alpha)$, and $\sin(\alpha)= -\sin(180+\alpha)$. Shown in Table 2-1, the second, third and fourth quadrants of a sine waveform can be constructed by using the phase to amplitude information of first quadrant $0\leq\alpha\leq90$. Just need to change the two MSBs of the output phase information of accumulator.

Table 2-1 Quadrant table of a sine wave in one period

Phase	MSB	MSB-1	Sine
0<A<90	1	0	Sin A
90<A<180	1	1	Sin(90-A)
180<A<270	0	0	-Sin A
270<A<360	0	1	-Sin(90-A)

Thus, given sin(X) over only the first quadrant, the operations necessary to flip to the other quadrants are as follows:

Quadrant I: *1/2+1/2 Sin X* *X is running index from 0 to* 2^{W-2}-1

Quadrant II: *1/2+1/2 Sin X* *X complemented (every bit inverted)*

Quadrant III: *1/2-1/2 Sin X* *X is running index from 0 to* 2^{W-2}-1

Quadrant IV: *1/2-1/2 Sin X* *X complemented*

The MSB determines the sign of the output and indicates whether the phase is in the first two quadrants or in the second. The MSB-1 is the quadrant bit and inverts the bits in the second and fourth quadrants

Figure 2-3 shows the implementation of a quadrant compressed sine ROM, The reduction of the ROM size is achieved because of the sine quadrant symmetry and because the inside ROM now maps W-2 input bits to D-1 output bits only, compared to W to D bits originally. The saving of ROM size is therefore

$$2^{W-2}\frac{D-1}{2^{W}D}=\frac{D-1}{4D}\approx 75\ saving \tag{2-21}$$

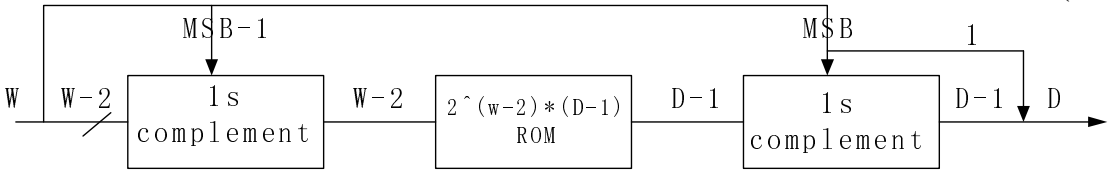

Figure 2-3 Quadrant compression of sine ROM

2.4 **Digital to Analog Converter**

The digital-to-analog converter (DAC) is the link that connects the digital signal generated, modulated, and manipulated in the direct digital synthesizer, and converts it into its analog form. And it is the parameter that has the greatest effect on DDS performance. In conventional ROM-based DDS, it's not easy to integrate in a small area and achieve output frequency bigger than 2GHz. The reason is the time and area consumed by ROM look up table. At present, ROM-less DDS becomes popular because it uses a nonlinear DAC to replace the ROM lookup table and the linear DAC and thus can generate much higher output frequency and occupy less area. [11][12].

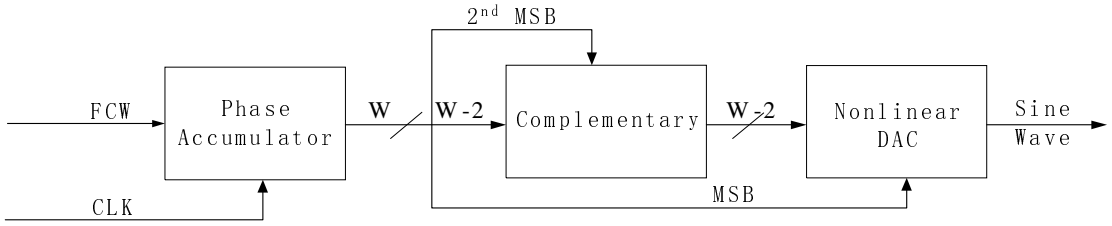

Figure 2-4 ROM-less DDS using nonlinear DAC

The function of a nonlinear DAC in DDS is to convert the linear phase information *W* at the output of the accumulator directly into analog sine waveform shown in Figure 2-4. Similar as conventional DDS, the truncated phase information *W* is divided into several parts. The first two MSBs are used to select different quadrants of the sine wave. The rest bits are segmented into m, n two parts. The first quadrant phase $0 \sim \pi/2$ is divided into $2^{(m+n)}$ phase steps equally. With the corresponding conversion, each phase step corresponds to a DAC output magnitude.

Generally, a switching matrix with row and column thermometer decoders is used to switch on and off each DAC cell according to the phase information. The DAC output is the sum of all the cells which's output of thermo-decoder turn on.

The first the quadrant sine wave form can be reconstructed by switching on and off different DAC cells with different values. For other quadrants of sine wave form can be built by flipping the phase and output waveforms as shown in Figure 2-3. The MSB output of the phase accumulator is used to provide the proper mirroring of the sine waveform at the π phase point. The second MSB is used to invert the remaining bits for the second and fourth quadrant of sine wave.

CHAPTER 3 DIGITAL MODULATIONS IMPLEMENTED IN DDS

3.1 Background

Nowadays, digital modulation is more popular than analog modulation in most modern wireless communication systems [13]. Actually it offers a number of advantages, such as increased channel capacity and the ability to transmit; receive information with higher accuracy than an analog communication system in the presence of noise and distortion.

On the other hand, the choice of digital modulation scheme will significantly affect the characteristics, performance and resulting physical realization of a communication system. It becomes a very attractive area. A DDS can implement various types of digital modulations since its circuitry can be used to program the frequency, phase and amplitude of a waveform. With the availability of single-chip high-speed DDS, such modulations can be done with high accuracy and at high speed.

The objective of this chapter is to discuss how to implement various phase and frequency modulations in DDS, plus how to filter the signal so as to reduce the transmitted bandwidth.

3.2 DDS Structure Used For Digital Modulations

Digital phase modulation generally adds a number of different phase angles to the carrier in a data period for different information. The simplest case is binary phase shift keying (BPSK). There are only two phase angles (0 deg or 180 deg) can be added to the carrier to represent logic zero or logic one respectively. Of course, we can also add

different phase angles to represent groups of bits. For example, byadding one of four different phase angles to the carrier (i.e., 45deg, 135 deg, 225 deg, and 315 deg) represents two-bit data at a time. This is called quadrature phase shift keying (QPSK). This idea can be extended further, eight different phase angles to represent three-bit data (8-PSK), or sixteen different phase angles to represent four-bit data (16-PSK).

On the other hand, instead of breaking the phase angle into discrete pieces, sometime we need to change the phase angle from one value to another smoothly. And this smooth, continuous phase modulation corresponds to frequency modulation. MSK and GMSK are the most popular ones.

Figure 3-1 [2] is the whole structure for digital modulations implemented in DDS. Fc is the carrier frequency, while Fd is the data rate.

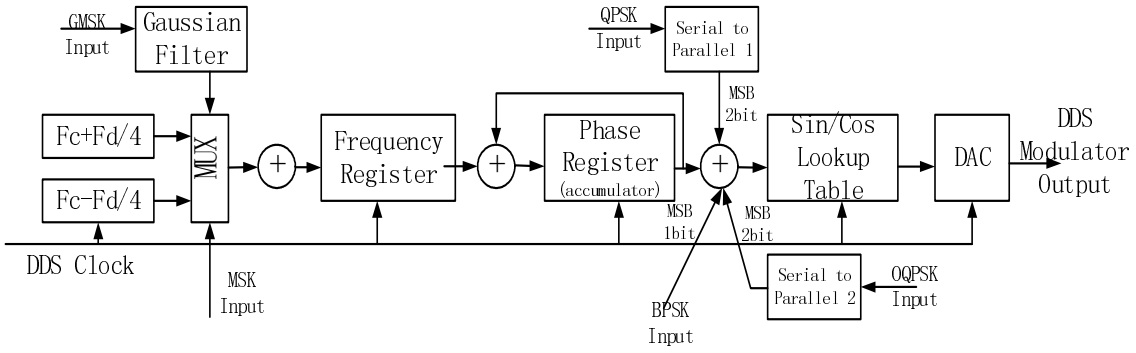

Figure 3-1 Direct modulation through a direct digital synthesizer

3.3 **Phase Modulation in DDS**

In order to transmit information, more than just a carrier tone is needed. The tone has to change in some way over time to indicate what information is being sent. In general, a carrier has two properties that can be changed or modulated in order to convey information to the receiver: amplitude and phase (frequency). Some modulation schemes only change the amplitude of the signal and some only change the phase (or frequency)

of the signal, but some more complicated modulation schemes change both. Here we only implement and verify these modulations changing one parameter. Let us begin by considering modulation that changes only the phase of the carrier.

3.3.1 BPSK & DBPSK

The simplest form of digital phase modulation is BPSK. In this case, when data is 0, the waveform has no change; when it becomes 1, the phase inverses 180 degree. Then we can write the carrier as

$$s_i(t) = A \times \cos[\omega t + \varphi_i(t)] \tag{3-1}$$

where the instantaneous phase angle, φi(t), is given by

$$\varphi_i(t) = \pi \times i \; ; \quad i = 0,1 \tag{3-2}$$

A is the amplitude of the carrier. Logic "1" corresponds to a phase inversion of the carrier, while logic 0 corresponds to no phase inversion. BPSK is often referred to as an antipodal modulation, where each of the two binary waveforms is the negative of the other shown in figure3-2. Each inversion of the carrier causes a sharp transmitted spectrum in the time domain response. These transitions produce a very wide transmitted spectrum.

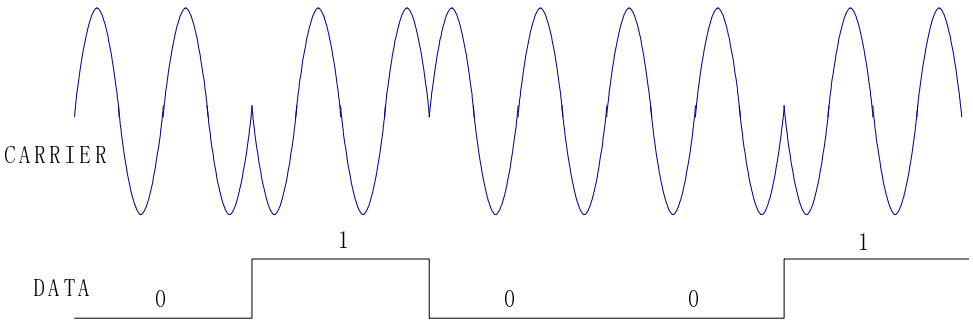

Figure 3-2 BPSK in time domain

Recall in Chapter 2, accumulator output represents the phase of sine wave and each bit has its own phase value.

$$\varphi = \frac{2\pi}{2^N} \times 2^{(N-n)} \tag{3-3}$$

where N is the accumulator's bit length and n is the accumulator's n^{th} bit (the left one is the 1^{st} and the right one is n^{th})

Thus, adding one to the highest MSB of accumulator equals adding 180 degrees to the present phase value; adding one to the second MSB equals adding 90 degrees to the phase value; for the third MSB, adding one equals adding 45 degrees to the phase value and etc. Generally, adding one to the n^{th} MSB of phase accumulator is equal to adding $\frac{180}{2^{(n-1)}}$ degrees to the carrier's present phase value.

As a result, to implement BPSK in DDS, we just need to add the data value to the accumulator's MSB (Figure 3-1). If the data is 1, the waveform change 180 degree in phase, inversely, if data is 0, it causes no phase transition. Consequently, data stream can be represented by the variety of waveform phase.

3.3.2 QPSK (Quadrature Phase shift keying)

Sometimes, more than one data bit needs to represent a symbol. Notice that, by using by using symbols to represent groups of several bits, we can reduce the rate at which the symbols are sent. In other words, if our bit rate is 2Mbps, then by using symbol to represent groups of 2 bits each, the signal rate is one-half the bit rate, or 1Mbps.

When we are doing different phase modulation, we might choose different phase angles for different data bits. In QPSK, each group of two bits (one symbol) is

represented by a different phase angles, 90 degrees apart from each other (0^o, 90^o, 180^o, 270^o). The phase can be changed ±90 deg or 180 deg. We can write the QPSK signal as

$$s_i(t) = A \times \cos[\omega t + \varphi_i(t)] \qquad (3\text{-}4)$$

where A is the carrier amplitude, i is an integer, and $\varphi_i(t)$ is the changeable phase angle of the modulated signal with respect to the un-modulated waveform.

$$\varphi_i(t) = \frac{\pi}{2} \times i \quad \text{where} \quad i = 0,1,2,3.$$

As mentioned earlier, each symbol represents a group of two bits. Therefore, the symbol rate of QPSK is half the bit rate of BPSK. This means that the variety of carrier is half rate of BPSK in the time domain response. The spectrum is only ½ as wide as that of BPSK for a given bit rate.

In order to implement QPSK in DDS, the only thing we need to do is adding proper data value to the highest two MSBs. Here we assign 0, 90, and 180, 270 degrees to represent 00, 01, 10, and 11 relatively.

Then another question comes, how to split one serial data stream into two parallel bit streams and each at one-half the rate of the original data? (Since two parallel bits convert to a phase change). Figure 3-3 shows the block diagram for a digital de-multiplexer that does this. The first two D-flip-flops take duty of alternating bits of serial data. The third one adds a 1-bit period delay to realign the two bit streams.

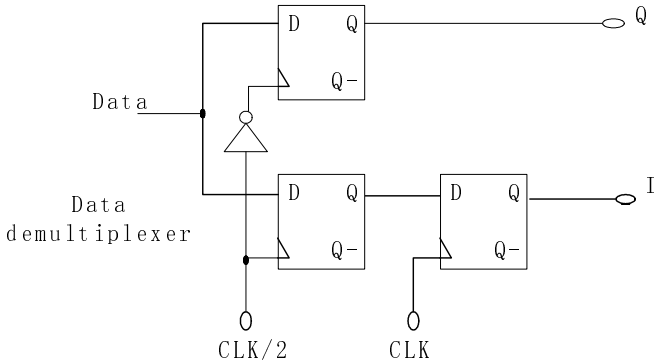

Figure 3-3 Data de-multiplexer for QPSK

Figure 3-4 shows the time domain response for QPSK. The phase of the carrier wave can be changed 0 degree, 90 degree, 180 degree and 270 degree.

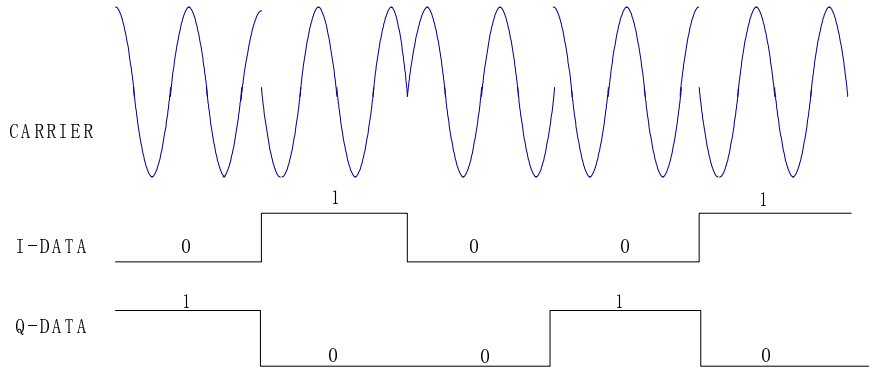

Figure 3-4 QPSK in time domain

3.3.3 OQPSK

Large amplitude variation in the QPSK waveform happened whenever the phase was changed by 180 deg. Sometimes, we might expect that if we could made the phase transition more gradual, then the QPSK waveform might be more tolerant to limiting. OQPSK is introduced for this purpose. In one symbol, there is only one bit changing, the other one remains. In order to implement this function, a new data de-multiplexer is shown in figure 3-5. We simply remove the third D-flip-flop, then there is a 1/2 bit delay between I data and Q data. Every time the symbol changes, only one bit changes. Consequently, there is no 180 degree phase transition, only ±90 degree change.

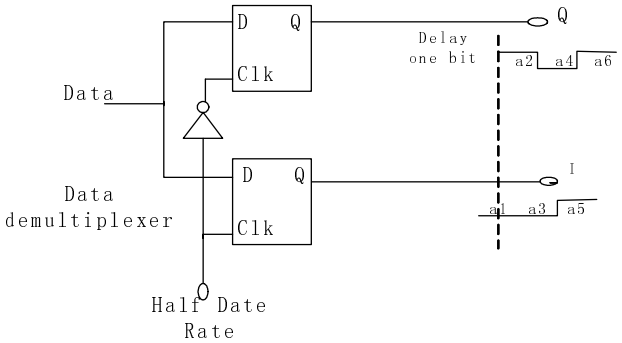

Figure 3-5 Data de-multiplexer for OQPSK

Figure 3-6 shows the time domain response for OQPSK. The phase of the carrier wave only changes ±90 degree. Here we assign 0, 90, and 180, 270 degrees to represent 00, 01, 11, and 10 relatively.

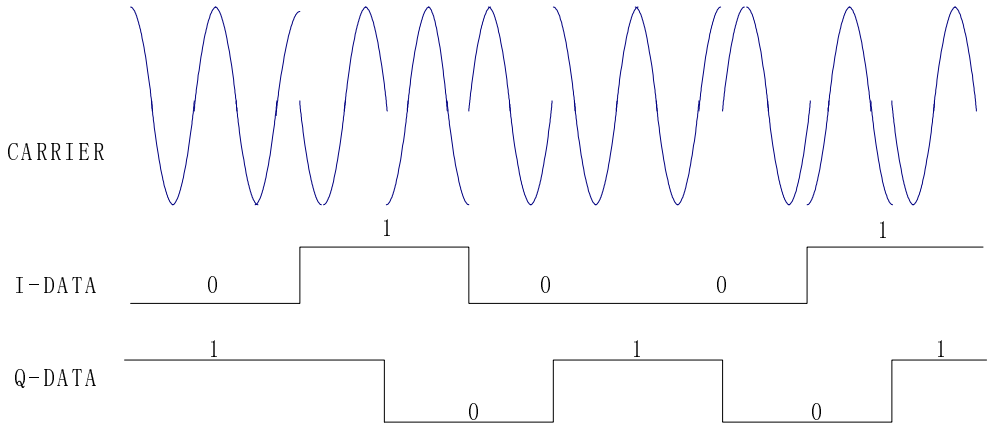

Figure 3-6 OQPSK in time domain

3.4 **Frequency Modulation in DDS**

Instead of breaking the phase angle into discrete pieces, sometimes we change the phase angle from one value to another smoothly (calls Frequency shift keying).

Recall that frequency is the derivation of phase

$$\omega(t) = \frac{\partial \varphi(t)}{\partial t} \tag{3-5}$$

Smooth, continuous phase modulation corresponds to frequency modulation. An FSK signal is given by:

$$S_{FSK}(t) = A \times \cos((\omega_{RF} + \omega_i)t + \varphi) \tag{3-6}$$

where A is the amplitude of the carrier signal (a constant), ω_{RF} is the nominal frequency of the carrier, ϕ is an arbitrary phase, and ω_i is the change in carrier frequency that determines what bits have been transmitted.

3.4.1 MSK (Minimum shift Keying)

In the MSK, the carrier gradually changes phase by ±90 deg over a bit period. Since the phase will continue to advance or retard itself over the course of each bit period, the frequency is varied. It can be represented by:

The frequency difference due to the input data value is shown below [14].

$$\Delta f(+) = \frac{\partial \omega}{2\pi} = \frac{\partial \varphi}{\partial t} \times \frac{1}{2\pi} = \frac{\pi}{2T_d} \times \frac{1}{2\pi} = \frac{1}{4T_d} = \frac{f_d}{4} \tag{3-7}$$

$$\Delta f(-) = \frac{\partial \omega}{2\pi} = \frac{\partial \varphi}{\partial t} \times \frac{1}{2\pi} = \frac{-\pi}{2T_d} \times \frac{1}{2\pi} = \frac{-1}{4T_d} = \frac{-f_d}{4} \tag{3-8}$$

$f_d = \frac{1}{T_d}$, f_d is the bit rate and Td is the bit period.

The frequency difference:

$$\Delta f(+) - \Delta f(-) = \frac{f_d}{2} \tag{3-9}$$

We can see in MSK the frequency difference that corresponds to a 90-deg phase advance or lag over a bit period is equal to one-half the bit rate. Therefore MSK can be easily implemented in a DDS -- to use MSK input data to toggle two frequency words corresponding to $F_c \pm F_d/4$, where F_c is the carrier frequency and F_d is the input MSK symbol data rate (shown in Figure3-1). When MSK data is zero, the DDS Frequency Control Word is given by $F_c - F_d/4$, which causes a continuous phase retardation of 90° within one bit period $1/F_d$. When MSK data is 1, the DDS output frequency is given by $F_c + F_d/4$, which causes a continuous phase advance of 90° within one bit period $1/F_d$.

As a result, the DDS output can be written as:

$$S_{\text{out}} = \sin\left(2\pi\left(F_C \pm \frac{F_b}{4}\right)t\right) = \cos\left(\pm\frac{\pi}{2}\cdot F_b t\right)\cdot\sin(2\pi\cdot F_C t) \pm \sin\left(\pm\frac{\pi}{2}\cdot F_b t\right)\cdot\cos(2\pi\cdot F_C t) \tag{3-10}$$

As shown, the DDS output is an MSK signal, which can also be thought of as OQPSK with sinusoidal pulse shaping on the base-band signal. When base-band MSK data is 1, the DDS FCW is given by $F_c + F_b/4$, which causes a continuous phase advance of 90° within one bit period $1/F_b$.

Figure 3-7 shows the time domain response for MSK. Here we assign $f_{carrier}=f_{data.}$ The frequency of the carrier toggles between two different frequency 3/4 $f_{carrier}$ and 5/4 $f_{carrier}$, according to the data value 0 and 1 respectively.

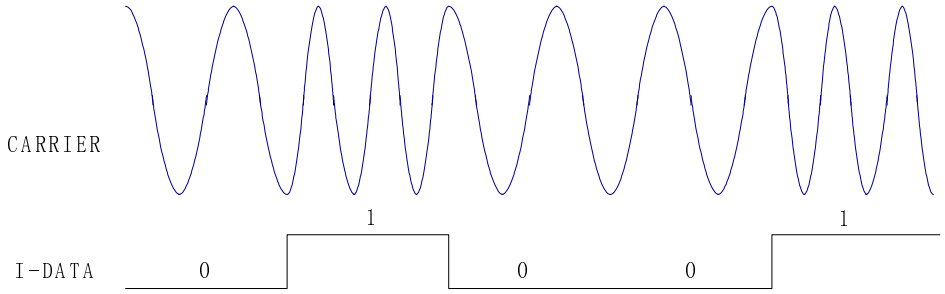

Figure 3-7 MSK in time domain

3.4.2 GMSK (Gaussian MSK)

On the other hand, the transmitted spectrum of MSK is sometimes too wide for many applications. One way to narrow the transmitted spectrum of MSK is by filtering the modulation signal before applying it to the DDS. This can be done by passing the modulation signal through a low-pass filter. Also, the filter needs to have a well-behaved time domain response. Therefore Gaussian filter is used before the modulators.

In this chapter we use FIR architecture to implement the Gaussian Filter. Fundamentally, an FIR is a very simple structure. There is nothing more than a chain of delay, multiply, and add stages. Each stage consists of an input & output data path and a fixed coefficient (a number which serves as one of the multiplicands in the multiplier section). It is told that increasing N will increase the overall delay through the FIR [15].

This can be a problem in systems that are not delay tolerant. However, there is a distinct advantage to increasing N; it increases the sharpness of the filter response.

Examination of the FIR structure in figure 3-8 leads to an equation for the output y(n), in terms of the input x(n) for an FIR of arbitrary length N. The result is:

$$y(n) = a_0 x(n) + a_1 x(n-1) + a_2 x(n-2) + ... + a_{N-1} x(n-N-1) \qquad (3\text{-}11)$$

(a_i is the Gaussian coefficient)

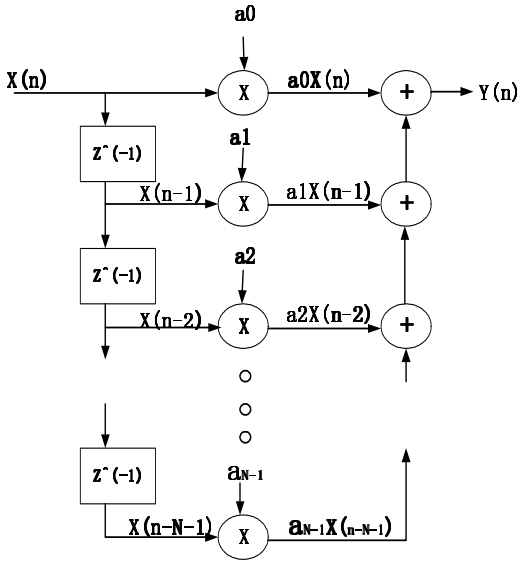

Figure 3-8 FIR filter structure

We use following equations to determine the Gaussian coefficient

$$h(t) = \frac{\sqrt{\pi}}{\alpha} \exp(-\frac{\pi^2 t^2}{\alpha^2}) \qquad (3\text{-}12)$$

$$\alpha = \frac{\sqrt{\log 2}}{\sqrt{2}B} \qquad (3\text{-}13)$$

Here B is the filter's 3-dB bandwidth. The BT product parameter is B times the input signal's symbol period. We let it be 0.5 in our design.

Examination of the above figure leads to an equation for the output, y(n), in terms of the input, x(n) for an FIR of arbitrary length, N. The result is:

$$y(n) = a_0 x(n) + a_1 x(n-1) + a_2 x(n-2) + ... + a_{N-1} x(n-N-1) \qquad (3\text{-}14)$$

Application of the z-transform leads to the transfer function, H(z), of an N-tap FIR filter.

$$H(z) = a_0 + a_1 z^{-1} + a_2 z^{-2} + ... + a_{N-1} z^{-(N-1)} \qquad (3\text{-}15)$$

Earlier, it was mentioned that increasing the number of taps in an FIR increases the sharpness of the filter response. Here, we give two instances which are plots of a 10-tap & a 2-tap FIR. All coefficients are equal to 0.1.

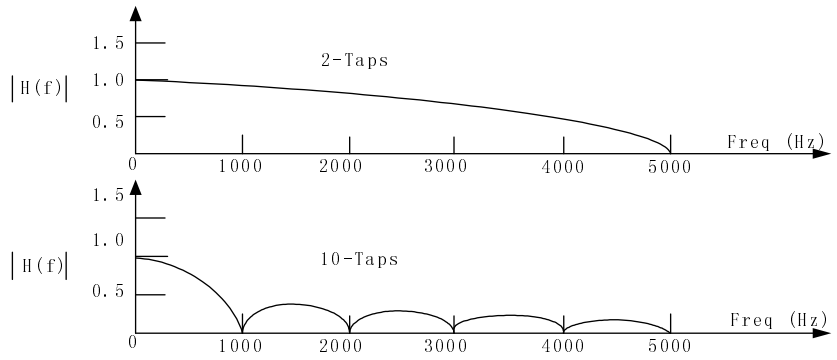

Figure 3-9 FIR with different taps

The main null ends at 1kHz for the 10-tap FIR, instead of at 5kHz for the 2-tap FIR. Another point should be mentioned, that the response of the FIR is completely determined by the coefficient values. By choosing the appropriate coefficients it is possible to design filters with just about any response; low-pass, high-pass, band-pass, band-reject, all-pass, and more [16].

Figure 3-10, Figure 3-11, Figure 3-12 shows a design example of a 16 taps Gaussian filters. The input data is NRZ with a rate at 3MHz (bit period T=$\frac{1}{3\times10^6}$). And this Gaussian filter's 3-dB bandwidth B=600 KHz; Thus BT=0.2.

Figure 3-10 shows the Gaussian function in time and frequency domain; Figure3-11 shows the input data in time and frequency domain; and Figure3-12 shows the data in time and frequency domain, which has went through the Gaussian filter. From these figures, we can see that, the Gaussian filter obviously minished the signal's frequency spectrum without bit errors.

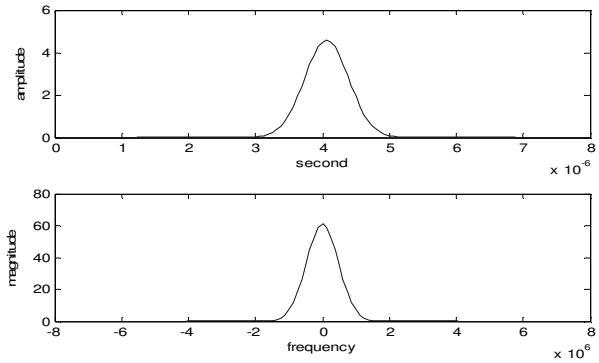

Figure 3-10 Gaussian waveform with 16 taps

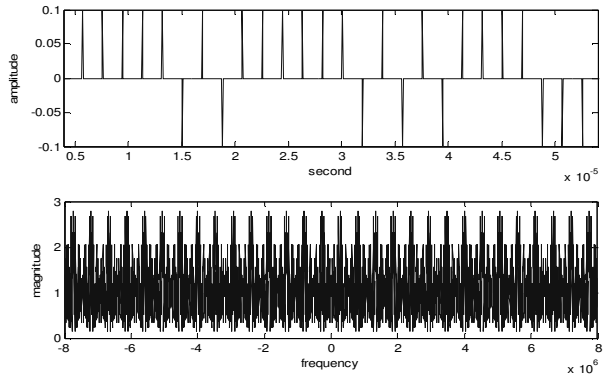

Figure 3-11 Input data

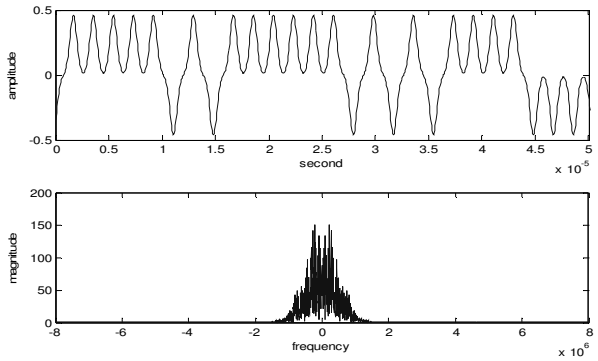

Figure 3-12 Gaussian filter's output

3.5 **Test Results**

In this digital modulation verification, AD7303 (provided by AD company) is used. It is a dual, 8-bit voltage out DAC that operates from a single +2.7 V to +5.5 V supply. Its on-chip precision output buffers allow the DAC outputs to swing rail to rail. And all the digital functionality was implemented in a filed programmable gate array (FPGA) board provided by Digilent Company.

A design example is given below with these parameters: bit length of phase accumulator =10, bit length of sine look-up table =10, bit length of DAC output =8.

3.5.1 BPSK

The carrier frequency is 48.8 KHz and the data rate is 32.6 KHz. The Data stream is 0 1 1 1 1 0 0, so carrier sine wave will add 180 degree at second data period and remove 180 degree at fifth data period. Each inversion of the carrier causes a sharp transition in the time domain response.

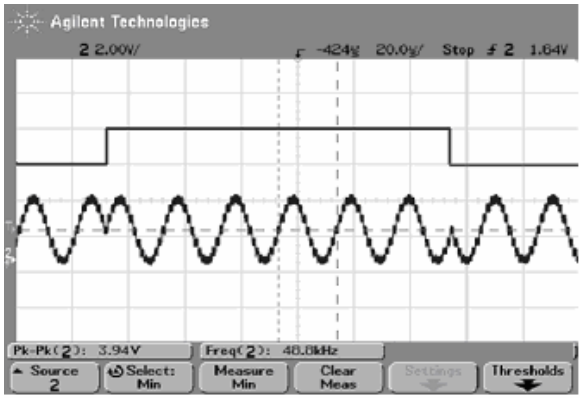

Figure 3-13 BPSK wave

3.5.2 QPSK

The carrier frequency is 48.8 KHz and the data rate is 65.6 KHz. Data stream is 0 1 1 0 1 0 1 0 0 0 0 0 0 0 0 01. After the de-multiplexer, the data split into two stream and each two bits represents one phase variation. The symbols 01, 10, 10, 10, 00, 00, 00, 01

represent the output phase changes 180 degree, 0 degree, 0 degree, -90 degree, 0 degree, 0 degree, 90 degree respectively.

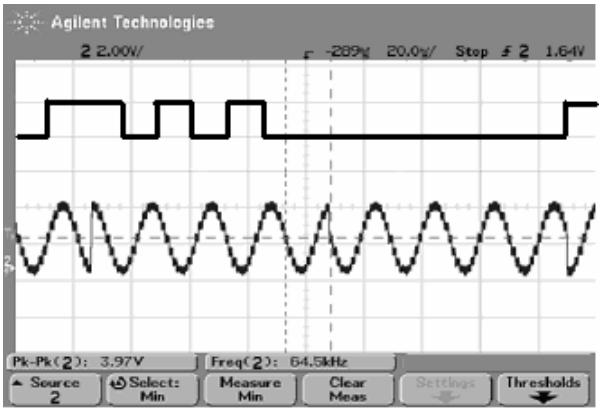

Figure 3-14 QPSK wave

3.5.3 OQPSK

The carrier frequency is 48.8 KHz and the data rate is 65.6 KHz. Data stream is 0 1 1 1 1 1 1 1 0 1 0 0 0 0 0. After the de-multiplexer, the data split into two stream and each two bits represents one phase variation. Since there is a half bit delay of one stream, there is only one bit changing each time. The symbols 01, 11, 11, 11, 10, 10, 00, 00 represent the output phase changes -90 degree, 0 degree, 0 degree, 90 degree, 0 degree, 90 degree, 0 degree respectively. We noticed that the largest phase change is 90 degree.

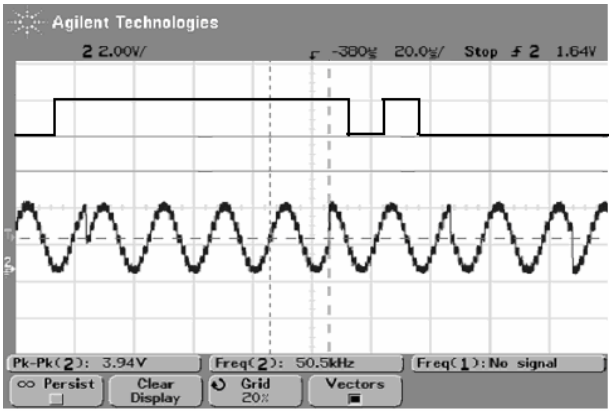

Figure 3-15 OQPSK wave

3.5.4 MSK

The carrier frequency is 44 KHz and the data rate is 48 KHz. Therefore, $F_C + \frac{1}{4}F_b = 56\,\text{KHz}$ and $F_C - \frac{1}{4}F_b = 32\,\text{KHz}$. Data stream: 0 0 1 0 1 0; the output frequency will be 32kHz, 32kHz, 56kHz, 32kHz, 56kHz, 32kHz respectively, during each period new data bit transfers.

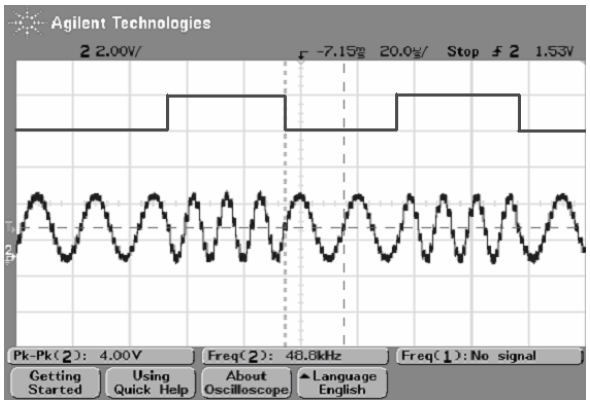

Figure 3-16 MSK wave

3.6**Conclusion**

Effective architecture used for BPSK, QPSK, OQPSK, MSK and GMSK have been proposed. The implementation was simplified by the use of DDS block, due to its own characteristics, which own high operation speed and high accuracy. Besides, FIR filter is discussed for narrow spectrum in GMSK.

From the test and simulation results it is clear that the introduced DDS modulation structure achieved our design goal. It goes without saying that this structure is good for reuse. The future research goal will be in terms of comparing the spectrums, bit errors, inter-symbol interference among these modulations, and make them meet the specific requirements in applications.

CHAPTER 4 PHASE-LOCKED LOOP OPERATION THERORY

4.1 **Introduction**

In semiconductor industry, phase locked-loop has a long history and a wide range of applications. PLL-based synthesizers are among the most common ways to implement synthesizers, and this filed attracts a great deal of attention and interest.

Different from DDS that generates the output frequency directly by setting input frequency control words, PLL is a feedback system that causes a particular system to track with another one. More detailed, we embed oscillators in a synthesizer environment so as to synchronize the output signal with a reference or input signal in frequency as well as phase. The comparing block in the circuit is commonly called a phase frequency detector. Actually, in a locked state, the frequency and phase errors between the oscillator's output signal and the reference signal are zero. The operation mechanism working on the PLL will clear up the phase error once it finds there exists one. As a result, the output signal is really locked to the reference signal. A typical PLL structure is shown in Figure 4-1 [17]. It includes phase/frequency detector, loop filter, oscillator and frequency divider.

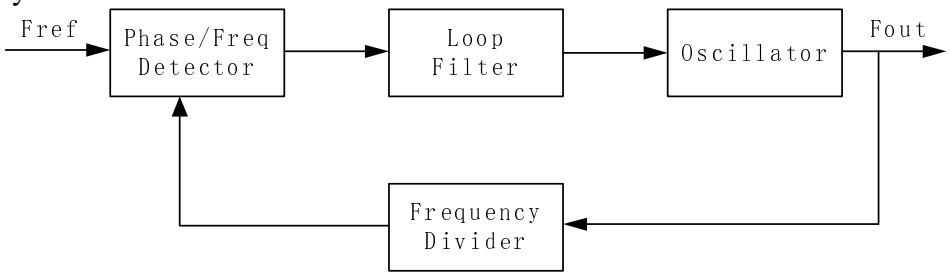

Figure 4-1 Typical PLL frequency synthesizer

4.2 **Integer-N PLL Synthesizer**

Synthesizers often require that the output frequency of a PLL be a multiple of the input frequency. As shown in Figure 4-1, to amplify the input, the output signal is divided down before it is fed back. Here, the N called the "modulus" refers to the frequency divider's divide ratio in the feedback circuit. Therefore, at locked state, the output frequency, which is a multiple of the feedback signal, is given by

$$f_{out} = N \times f_{ref} \tag{4-1}$$

As we know, the frequency synthesizer generates an output frequency can be given by $f_{out} = f_0 + kf_{ch}$, where f_0 is the lower end of the range, k is an integer varying from 0 to the maximum number of channels and the f_{ch} is the frequency step. Consider with (4-1), the $N \times f_{out}$ is to be equal to $f_0 + kf_{ch}$, where N varies in unity step from N_L to N_H. Then, for the first channel (k=0), we have $N_L \times f_{ref} = f_0$; for the second channel, $(N_L + 1) \times f_{ref} = f_0 + f_{ch}$, implying that $f_{ch} = f_{ref}$. The important point here is that the input reference frequency must be equal to the channel spacing. As a result, in order to get a smaller step size, the reference frequency must be made smaller.

4.3 **Charge Pump PLL**

Charge pump PLL frequency synthesizer is widely used for its simple structure and good performance. In the loop filter considered in Figure 4-1, the average value of the phase detector output is obtained by depositing charge on to a capacitor during each phase comparison and allowing the charge to decay afterwards. On the other hand, if a phase detector is connected to a charge pump, there is negligible decay of charge between phase comparison instants. A typical charge pump PLL is composed of a phase detector, charge pump, loop filer and VCO as shown in figure 4-2.

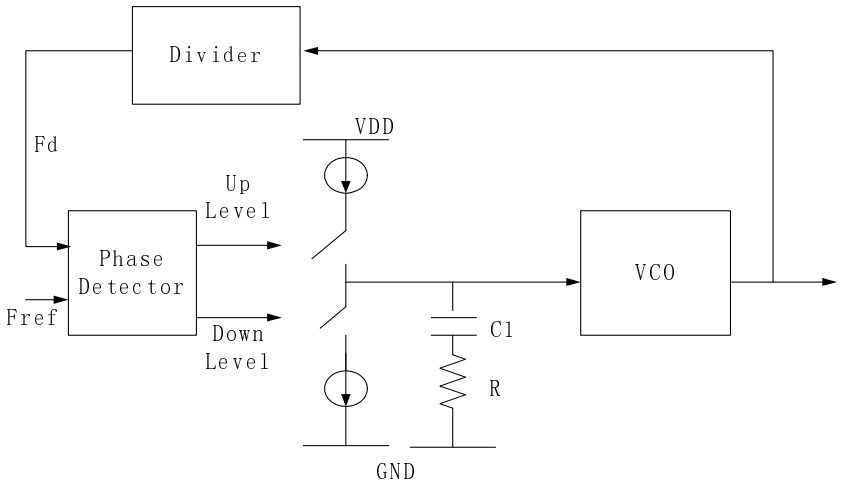

Figure 4-2 Charge pump PLL

The charge pump PLL works as follow. The pump itself consists of two switched current sources driving a capacitor. The current sources controlled by the outputs of PFD charge and discharge the loop filter capacitors to generate the voltage control signal to VCO. Phase detector transforms the phase or frequency error signal to two switching signal UP and DOWN with corresponding pulse width. If the fref > fd, the positive charge accumulates on C1 steadily. Similarly, if the fref< fd, the I2 removes charge from C1 on every phase comparison. If the output of phase detector is 0, the voltage on capacitor remains constant.

Compared with conventional PLL, charge pump PLL has three main advantages: (1) The capture range is only controlled by the VCO output frequency range; (2) If the mismatches and offsets of the two branches currents are neglected, the static phase error is zero; and (3) The certain amount of time to lock on the frequency of an incoming signal is short and the synthesizer's switching speed is fast. Moreover, the open-loop transfer function of it has two poles at the origin and root locus shows that more stability of the PLL is achieved with the increasing loop gain.

The whole loop dynamics can be analyzed by the linearized mode of each component in the PLL. The PFD has a gain of K_{PFD}, the loop filter has a transfer function F(s), and the VCO has a gain of Kvco/s. Suppose the loop begins a phase error ϕ_{ref}-ϕ_d=ϕ_e. The average current charging the capacitor is given by $I\phi_e/(2\pi)$. The average change in the control voltage after the loop filter is:

$$V_{ctrl}(s) = \frac{I\phi_e}{2\pi}(R + 1/sC_1) \tag{4.2}$$

The output phase from VCO is:

$$\phi_{out}(s) = V_{ctrl}(s)Kvco/s \tag{4.3}$$

We obtain the following closed-loop transfer function:

$$H(s) = \frac{\phi_{out}(s)}{1+\phi_{out}(s)} = \frac{I \cdot Kvco(RC_1 s + 1)/(2\pi C_1)}{s^2 + I \cdot Kvco \cdot R \cdot s/(2\pi) + I \cdot Kvco/(2\pi C_1)} \tag{4.4}$$

The loop transfer function has a zero at ω_z=-1/(RC_1). And the second order system has two important parameters, ω_n is proportional to loop bandwidth and independent of R;

$$\omega_n = \sqrt{I \cdot Kvco/(2\pi \cdot C_1)} \tag{4.5}$$

$$\zeta = (R/2) \cdot \sqrt{I \cdot C_1 \cdot Kvco/(2\pi)} \tag{4.6}$$

Suppose the divider modulus changes form M to N+k and k<<N, the frequency settling time inside the loop is:

$$t_s \approx (1/\zeta \cdot \omega_n)\ln(k/(N|\alpha|\sqrt{1-\zeta^2}) \tag{4.7}$$

Note that the decay time constant of the system is equal to $(1/\zeta \cdot \omega_n)=4\pi/(R \cdot I \cdot Kvco)$ which is independent of C_1.

In many cases, how to maximize the loop bandwidth and shorten the settling time is highly desired. This can be achieved by increasing the pump current I and VCO's coefficient Kvco as shown in (4-4). However, as the loop bandwidth becomes comparable with the input reference frequency, the loop would become unstable. Usually, loop bandwidth ω_n should meet equation (4.7) for the stability of the loop *d1* [18].

$$\omega_n^{\,2} < \omega_{ref}^{\,2} / (RC_1 \omega_{ref} \pi + \pi^2) \tag{4.8}$$

In typical design, the loop bandwidth is roughly one-tenth of the input frequency to guarantee stability.

The loop filter in figure 4-2 indicates that the loop is a second order system. In practice, another capacitor C_2 is added in parallel to the R and C_1. C_1 produces the first pole for the loop filter and C_2 is used to smooth the control voltage ripples and to generate the second pole. This modification leads the system to a third order system.

4.4 **Fractional-N Frequency Synthesizer**

In the integer-N architecture, the loop bandwidth is limited and the input reference must be equal to the channel spacing. This is often undesirable, so instead, a fractional-N design is often used. Instead of an integer-N divider, we put a fractional frequency divider in the feedback of PLL, shown in Figure 4-3. It allows the PLL to operate with high reference frequency while achieving a small step size. The fractional divider is usually a multi-modulus divider (MMD). A modulus control signal is used to toggle different modulus at each reference period.

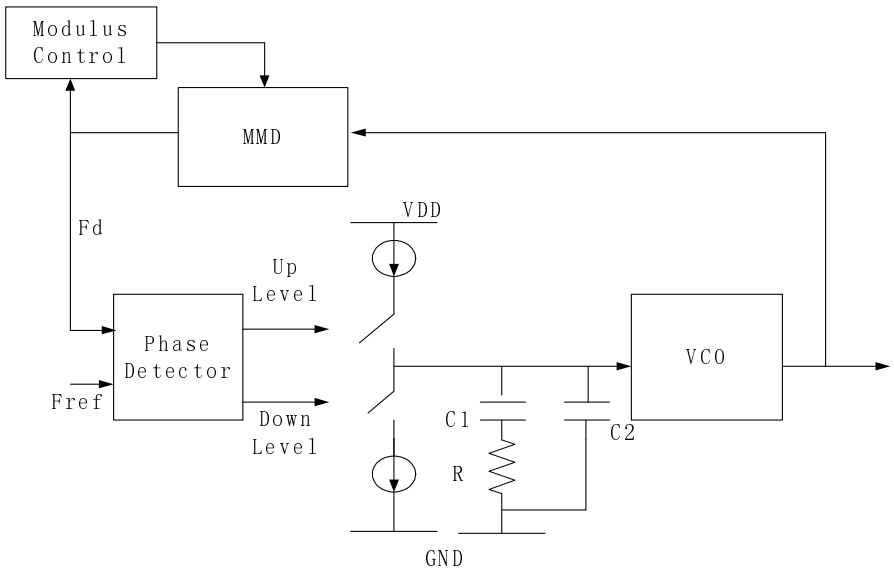

Figure 4-3 Fractional-N PLL frequency synthesizer

For a two modulus divider case, the divider can divide the input signal from VCO by *N* or *N*+1. Assume that the divider divide By *N* for *A* reference cycles and *N*+1 for *B* reference clock cycles. The N_{avg} is defined as the average division ratio:

$$N_{avg} = (A+B)/[A/N + B/(N+1)] \tag{4-9}$$

This value can vary between N and N+1 in fine steps by proper choice of A and B. Thus the frequency step can be a fractional of reference signal and fractional-N synthesizer can have a higher reference frequency and higher loop bandwidth without stability problem. For example, with the reference frequency in the range of tens of megahertz, the loop bandwidth of a fractional-N synthesizer can be as high as a few megahertz, yielding a fast settling time of the PLL and suppressing the noise contributed by the VCO. Furthermore, the smaller division ratio lower the effect contributed by the reference source, the PFD, the charge pump noise [19] [20].

Replacing the dual-modulus divider with a MMD in PLL can be modified to a more generic form. It is possible to have a much smaller frequency step with the same

reference frequency comparing to the traditional PLL synthesizer. The synthesizer output frequency is given by [2].

$$f_o = \frac{f_r}{R}\left[I + \frac{K}{F} \right] \tag{4-10}$$

where I is the integer portion of the frequency divider and, depending on the complexity of the design, I could have many possible integer values. For example, if loop division ratio 200.75 is needed, we can program I=200, K=3 and F=4. The MMD division ratio is toggled between 200 and 201.

A popular MMD topology used cascaded 2/3 cells. With an n-bit (P1~Pn) modulus control signal, the MMD division ratio is shown as:

$$N = P_1 + 2^1 P_2 + ... 2^{n-2} P_{n-1} + 2^{n-1} P_n + 2^n \tag{4-11}$$

4.5 **Fractional Spurs**

The alternation modulus of the divider cause the output frequency can vary among different division ratios. However, it suffers from a common side effect of generating spurious components. As we know, any repeated pattern in the time domain causes spurious tones in the frequency domain. The fractional accumulator periodically generates the carry-out that toggles the loop division ratio. And if this frequency falls inside the loop bandwidth, large spurious tones are expected. One way to suppress these tones is have a very small PLL bandwidth, which negates the potential benefit of the fractional-N technique.

Since the phase difference between the feedback and reference signals grows to significant values, the amplitude of loop filter output waveform is quite large, yielding fractional spurs only 20dB below the carrier magnitude [1]. Consequently, several

methods introduced to suppress these spurs. One is known as the “fractional compensation”. If the divider's alternating modulus is deterministic, the current generated by the charge pump can be compensated by another current pulses that with the same alternating process but an opposite sign. This technique depends on highly accuracy of the compensation currents [21].

The other way to eliminate fractional spurs is to randomize the sequence of the modulus so as to break up their periodicity. With the randomness of the modulus, the average division modulus can still be achieved and the fractional spurs are converted into white noise. The white noise inside the PLL’s bandwidth is integrated by the PLL transfer function and introduces phase noise contribution no mater what PLL bandwidth is.

Delta-sigma fractional-N PLL can solve this problem by shaping the noise spectrum to higher frequency offsets [22] [23]. It generates the sequence of modulus such that the quantization noise has most of its power in a frequency band well above the desired bandwidth of PLL.

CHAPTER 5 RFIC DESIGNS FOR AN INTEGRATED PLL IN 0.5 UM SIGE TECHNOLOGY

5.1 Introduction

In semiconductor industry, Phase locked-loop has a wide range of applications. Also it is another very important frequency synthesizer in RF system.

In this chapter, we will discuss the RFIC design of each main block in a fractional-N PLL: reference buffer, phase frequency detector, charge pump bias current setting, programmable charge pump, LC-tuned VCO, programmable divider and multiple modular divider (off-chip loop filter is used). Among these blocks, the integrated VCO is very important one. Even though the wideband loop can suppress the noise from the VCO, the suppression may not be enough because the integrated VCO is noisy, and the loop bandwidth cannot go arbitrarily high. A careful designed VCO is crucial in achieving a high performance frequency synthesizer. On the other hand, MMD and programmable divider consume more than three-quarter of PLL power, need to run at very high operation speed and contribute to a significant part to the in-band phase noise, thus they should be also key blocks in phase lock loop. Then how to use less current and simple structure to achieve the high speed and less noise is our goal. Of course, phase frequency detector, charge pump, and loop filter are also important in realizing a high performance frequency synthesizer. Fig. 5.1 shows the fractional-N PLL block diagram

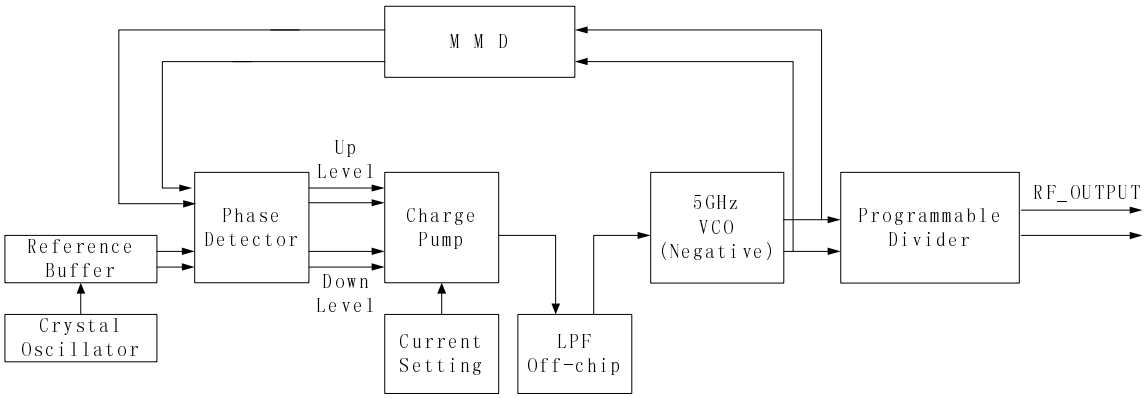

Figure 5-1 Block diagram of a PLL

5.2 **Current Mode Logic**

When we design low frequency logic circuits, CMOS rail to rail is the most commonly used type, for its DC power consumption is zero [24]. However, the integration of digital functions with sensitive RF signal processing blocks is hard to achieve when using the standard rail-to-rail logic techniques. The reason is the generation of large supply and substrate disturbances during logic transitions. Besides the rail to rail logic output changes by a large amount and requires larger charge and discharge time and can not achieve high speed operation. Therefore, the current mode logic (CML) has been introduced and employed in the high-speed digital systems for it very fast switching property.

The basis of all CML is the differential pair [25][26]. That means the input signal should have large enough voltage to switch the differential pair in digital applications. For bipolar, when the differential voltage is approximately four times of thermal voltage or larger, the input pair can be completely switched.

Normally, the CML circuit is composed of several differential pairs which are tied together either in series or parallel with a constant current source at the bottom. When the

circuit works, only one branch of each pair switches on at one time and the current is flowing through only one path from power supply to ground. Figure 5-2 shows a four level 3-input circuit. It is a three inputs AND gate. Here, one of the two resistors pulls down the output voltage from VCC to VCC-$I_{Bias}R_L$ when bias current flowing through it. Then the output differential swing is thus $2I_{Bias}R_L$. If we want to keep the output swing, when the bias current is increase, the resistance of R_L should decrease to keep the product.

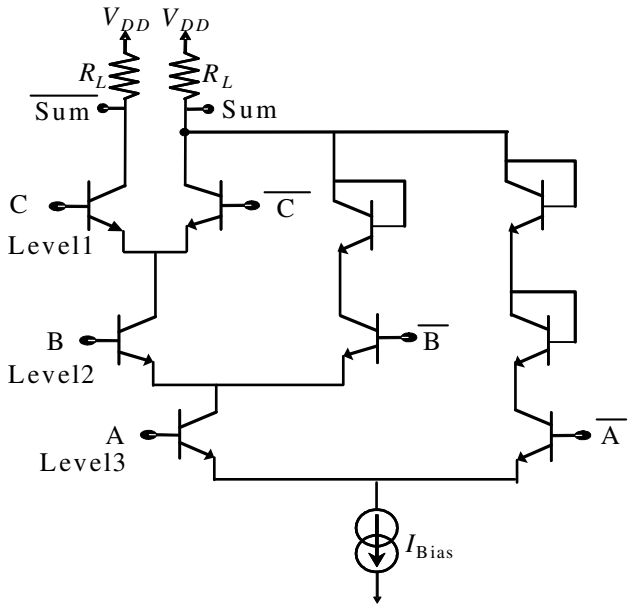

Figure 5-2 A CML circuit

In this research work, with a 3.3 V supply, CML makes it possible to have three level inputs which are bias at 3.2V, 2.3V and 1.4V so as to make sure the transistors not work in saturation mode. And these inputs share one constant current source, which would greatly save the power consumption. (Since more logic gates mean more current streams, when we use the CML structure to combine two logic functions together, the power consumption would be greatly saved.)

Generally speaking, with CML circuits, more current results faster operation. In order to achieve maximum speed, it's a good decision to make the bias current as large as possible by setting it to peak f_T. However, more current means more power consumption, and latter is also a very important specification in RF IC design [27][28]. Thus, we need to make a good choice to balance these two.

5.3 Multi-Modular Divider with Extension

Synthesizers often require that the output frequency of a phase locked loop be a multiple of the input frequency. A PLL can "amplify" a frequency such as a feedback circuit amplifies a voltage. Since the output of a PLL is the frequency, a frequency divider must be inserted in the feedback loop, dividing the oscillator output down to the reference frequency. As PLL definition says when the loop is locked, $\omega_{ref} = \omega_{div,}$ and hence $\omega_f = \omega_{vco}/M$. If there is only one modular, the VCO output changes by only integer multiples of input reference frequency, thus the input reference must be equal to the channel spacing. Actually, sometimes the output frequency need vary by a fraction of the input frequency, which allows the latter to be much greater than the channel spacing. This kind of divider is dynamically switched among different divide ratios to generate the equivalent of a fractional-divider ratio.

5.3.1 Architecture

Actually, the oscillator drives the divider input and, since this is the highest frequency in the circuit, speed is a big challenge in divider design. Also in RF design, high speed requires big operation current. So power dissipation is a problem too. Here we discuss a multi-modular divider. In this case, at each subsequent stage, the speed is lower, and the

power dissipation is reduced. Fig 5-3 [29] shows the programmable MMD architecture we used in the design.

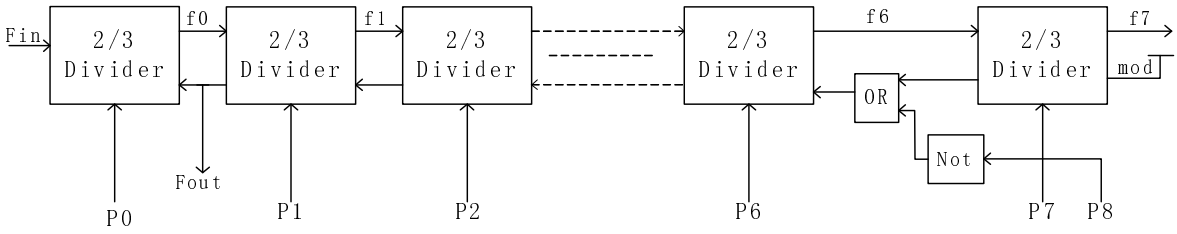

Figure 5-3 8 bit MMD architecture with extension

The modular structure consists of a chain of 2/3 divider cells and the feedback lines are only present between adjacent cells. This “local feedback” enables less operation delay and simple optimization of power dissipation. A D-flip-flop clocked by the VCO can be placed at the MMD output so that the noise from the divider does not contribute at the output of the PLL. The noise from the PFD, charge pump and loop filter is multiplied by the MMD’s divide-ratio at the output of the PLL.

We can briefly describe the MMD operation. Once in a division period, the last cell on the chain generates the signal mod_{n-1}. This signal combined with the programming input P8 then propagates “up” the chain, being re-clocked by each cell along the way. Pay attention here, no matter what value of $mod_{n-1,}$ when P8 is 0, the input mod signal of (n-1) block is always 1 without any division.

An active *mod* signal enables a cell to divide by 3, provided that its programming input p is set to 1. Division by 3 is adding one extra period of each cell’s input signal to the period of the output signal. Therefore, a chain of 2/3 cells provides an output signal with a period of

$$
\begin{aligned}
&Tout \\
&= 2^{n} Tin \times p_{n} + 2^{n-1} Tin \times p_{n} \times p_{n-1} + 2^{n-1} Tin + 2^{n-2} Tin \times p_{n-2} \\
&+ 2^{n-3} Tin \times p_{n-3} + ... + 2Tin \times p_{1} + Tin \times p_{0} \\
&= (2^{n} + 2^{n-1} p_{n-1}) \times Tin \times p_{n} + (2^{n-1} + 2^{n-2} \times p_{n-2} + ... + 2 \times p_{1} + p_{0}) \times Tin
\end{aligned}
\tag{5-1}
$$

In (4-1) *Tin* is the period of the input signal, and p_0~p_{n-1} are the binary programming values of the cells 1 to n respectively. P_n is the binary input control for extension. The equation shows that all integer division ratios ranging from (if all p = 0)2^{n-1} to 2^{n+1}-1 (if all p=1) can be realized. In our design, the n=8, so the divide ratio is from 128 to 511.

5.3.2 Dual-Modular Divider

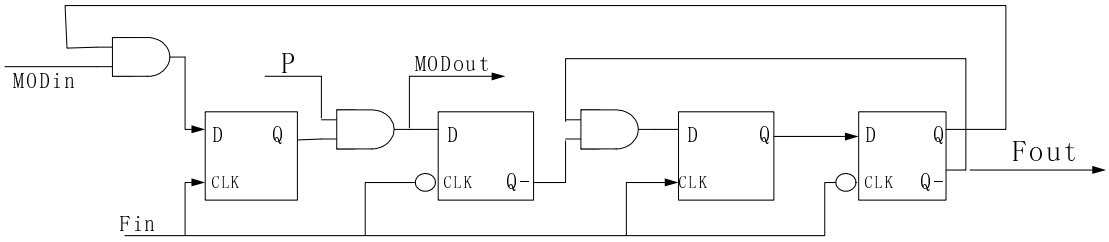

Figure 5-4 2/3 divider cell

A 2/3 divider cell comprises two functional blocks: AND gate and Latch (as depicted in Fig. 5-4) [30][31]. This block divides the frequency of the input signal either by 2 or by 3 and outputs the divided signal to the next cell in the chain as clock. The division ratio of the cell is based on the state of the mod_{in} and p signals. If the mod_{in} signal is 0, the divide ratio is always 2. When the mod_{in} signal becomes active once in a division cycle, the state of the p input is checked, and if p = 1, the end-of-cycle logic forces the pre-scaler to swallow one extra period of the input signal. In other words, the cell divides by 3. If p = 0, the cell stays in division by 2 mode. Regardless of the state of the p input, the end-of-cycle logic reclocks the mod_{in} signal, and outputs it to the preceding cell in the chain (mod_{out} signal).

5.3.3 Circuit Implementation of The 2/3 Divider Cell

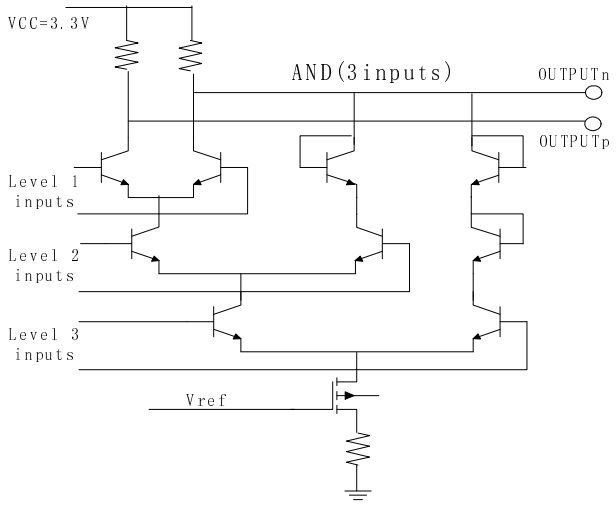

Figure 5-5 CML circuit design

In this circuit design, all digital blocks are using current-mode logic. The reason is the generation of large supply and substrate disturbances during logic transitions. However, the CML circuits, due to its constant supply current and relatively smaller differential voltage switching operation, have better EMC properties. Besides, CML can reach much higher operation frequency than rail-to-rail logic. The logic functions of the 2/3 cells are implemented with the CML structure presented in Fig 5-5. The logic three combines an AND gate with a latch function. Therefore, seven logic functions are achieved by four small blocks and at expense of four tail currents only.

The differential voltage swing is set to 400 mV and the three level inputs should be bias at 3.2V, 2.3V and 1.4V. The constant current source is set by current mirror with the same transistors and load resistances

5.3.4 Power Dissipation Optimization

As showing in Fig 5-3, the modular structure consists of a chain of 2/3 divider cells and the feedback lines are only present between adjacent cells. The absence of long delay

loops in the architecture enables fast and reliable optimization of power dissipation, since simulation runs may be done for clusters of two cells each time [32].

When going through the circuit, we can see that there is a maximum delay between the *mod* and the clock signals in a given cell that still allows properly timed division by 3. The maximum delay is $T_{max}=1.5 * T_{in}$ where T_{in} is the period of the cell's input previous one. As a consequence, the maximum allowed delay increases as one moves "down" the chain. As the delay in a cell is inverse proportional to the cell's current consumption (which is a property of current mode logic circuits), the currents in the cells may be scaled down as well. For high frequency, we use big current to operate; when it comes to lower frequency, we can use less current to drive. On the other hand, the input signal is not shared by all cells. It is only used for the last stage. Others use the outputs of anterior stages. In this case, there is no need to build a 'clock tree' to drive all blocks.

When a D-flip-flop clocked by the VCO is added at the output of the MMD, the noise generated by the divider itself is bypassed Simulation of extracted MMD shows in Figure 5-6. Input clock = 150p (6.7GHz), Division ratio 128; Output signal of MMD=19.2n (52.1MHz)

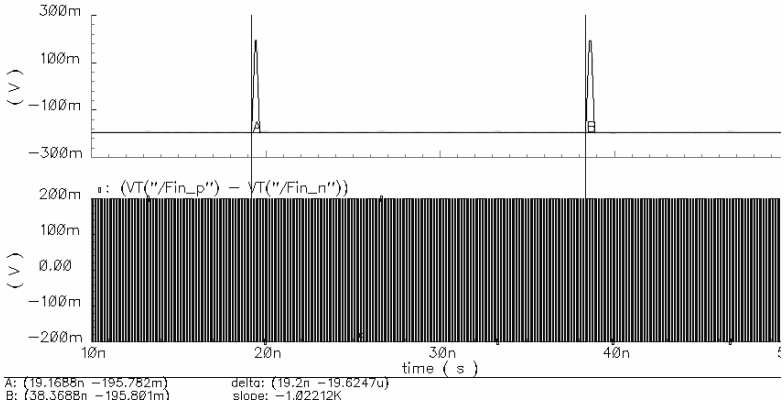

Figure 5-6 MMD simulation

5.4 **Programmable Divider (divide ratio 1~8)**

The wireless communication market has greatly expanded to many aspects. And trends are moving towards lighter weight, smaller size, longer lifetime and higher operation speed. On the other hand, due to the presence of a variety of standards covering different applications and employing a wide range of frequencies, having one portable device working for several standards with low power and low cost is highly desirable.

The easiest way is to add a high speed programmable divider following the VCO. It is an area consuming solution for having several synthesizers each one designed for a specific standard.

In ours circuit, we design a programmable divider whose modulus can be varied from 1 to 8. Programmability is achieved by sending the output signal back to the first D-flip-flop after the signal going through the new "MUX". Based on this structure, the modulus can be extended or changed depending on how many control bits and D-flip-flops to be chosen.

5.4.1 Divider's Architecture

Figure 5-7 shows the divider architecture based on a MUX and a chain of DFFs. We first consider a simple $\div 1$ status (when control bits ABC=000). The MUX chooses the clock signal and feed it back to the first DFF input. After the signal goes though the DFF chain, there is no frequency difference between the input and output except time delay. When the control bits (ABC) become 001, the input frequency is divided by 2. In this case, MUX chooses Q1- to feed back to the first DFF input. Therefore, the last three DFFs do nothing with the division ratio but time delay, only the first DFF output (Q1-)

contributes to the division ratio [1]. And if the control bits change to 010, MUX chooses combination of Q1-Q2-. The divider employs first two DFFs together with an AND gate to create three states (Q1-Q2- =01, 10, 11, but no 00) , and the last two DFFs are only for time delay. For other division ratios, operational mechanism is very similar.

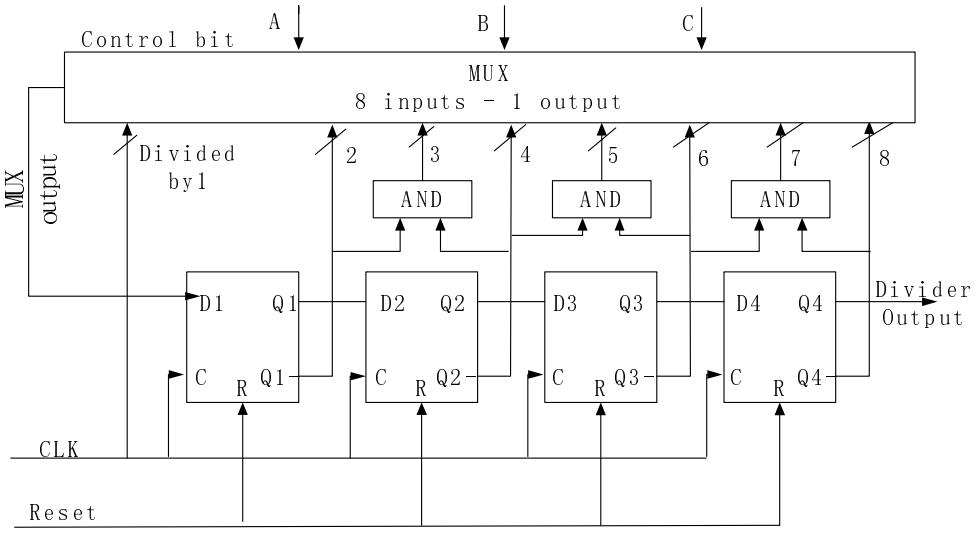

Figure 5-7 Programmable divider architecture

Meanwhile, we notice that the reason of propagation delay is the feedback circuit (MUX) and the DFF chain. Since the DFF chain's delay cannot make logic errors, MUX is the key block that determines the operation speed.

Traditionally, there are two ways to implement multiplexer: one is composed of simple 2-1 MUX, shown in Figure 5-8 [33]. ABC determine which signal (D1~D8) goes through these 2-1 MUX circuits. The shortest signal path has three logic gates in series. Another structure built by AND, OR gates and decoder is shown in Figure 5-9. In this architecture, input signals do not go through the decoder circuit, so the decoder does not contribute to the propagation delay. However, there are still at least three logic gates in the shortest signal path. More logic gates in series means more propagation delays. It

greatly affects the operation speed. Therefore, how to decrease the number of logic gates in signal path is the key point of operation speed. Besides, the more logic gates are used, the more chip area and power consumption would take.

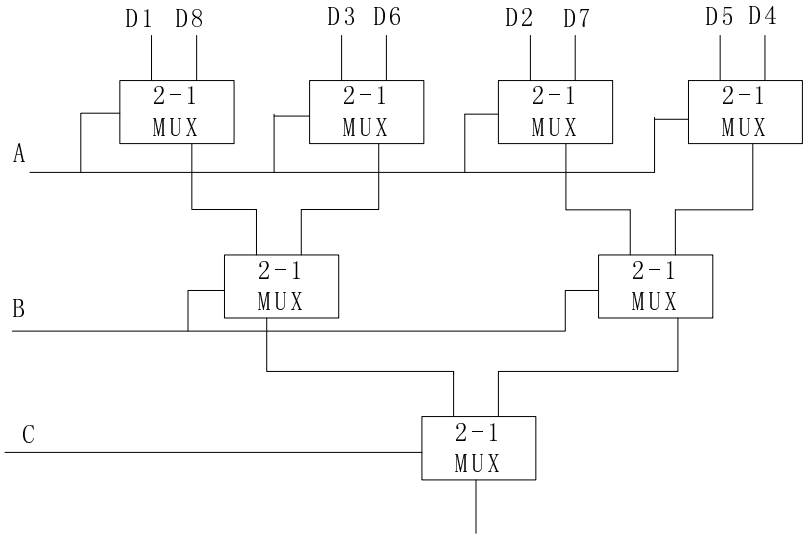

Figure 5-8 8:1 MUX composed by 2:1 MUX.

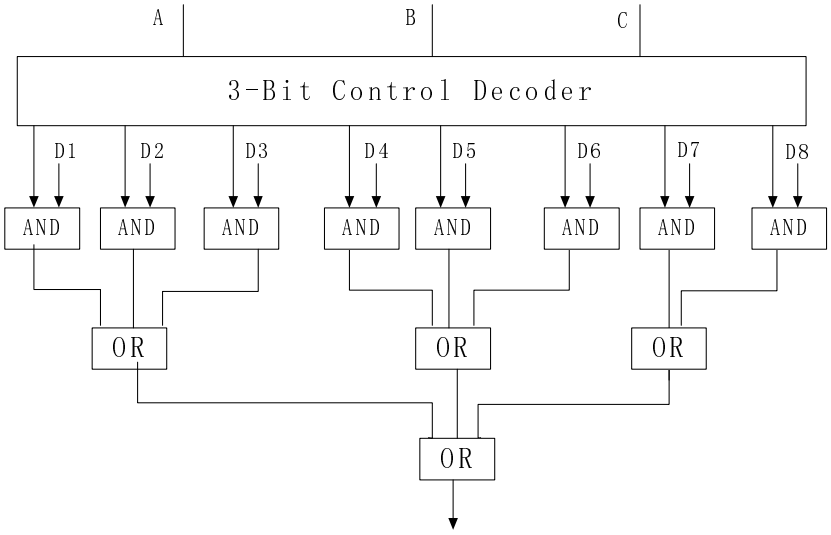

Figure 5-9 8:1 MUX composed with decoder and logic gate.

5.4.2 Logic Implementation of Digital Blocks

1) Logic Implementation of 3 Bit Decoder

The schematic of this control block is given in figure 5-10. It consists of three-input CMOS-CML gates. Then the three single inputs will change to three differential pairs

of inputs. This differential characteristic of CML makes it very easy to achieve additional "NOT" logic, just need to exchange positive and negative signals at inputs. As we know only if the all inputs of the AND gate are one then output would be one, otherwise it is zero. We use this principle for our decoder design. For 3-control-bit, we can get 8 different logic combinations with using AND gates (A-*B-*C-), (A-*B-*C), (A-*B*C-), (A-*B*C), (A*B-*C-), (A*B-*C) (A*B*C-), (A*B*C). At each time, only one combination will be logic one.

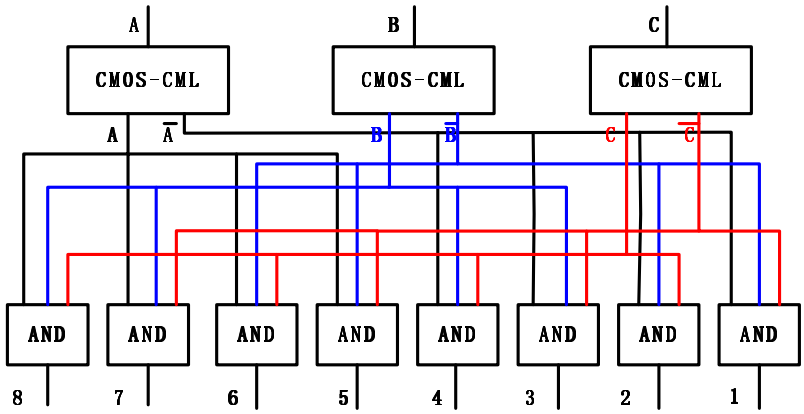

Figure 5-10 3-bit decoder (8 outputs)

2) Logic Implementation of Low-Power MUX

The MUX presented in this paper takes advantage of CML characteristic -- the whole circuit shares only one constant current source. Figure 5-11 shows the details.

MUX's first and second level inputs connected to the DFF outputs and MUX's outputs connected to the first DFF inputs. In channel 1, 2, 4, 6, 8, only first level has input signals, while second level put diodes there just for voltage bias. In channel 3, 5, 7, both first and second levels have input signals. And these two levels can be used as AND logic to combine the input signals. Here, we can save three AND gates shown in Figure 5-7.

The third level is the control level. We connected the decoder's outputs to the third level inputs. During the operation, decoder has only one output to be logic high, while others are logic zero [34]. For instance, when ABC=000, the input 1 is logic high and the upper signals can go through the channel as circuit outputs, while other channels are closed and there are no signals pass through; when ABC=001, the input 2 is logic high and the combination signal Q1-Q2- go through the channel and are selected as outputs, while other channels are closed, etc.. In other words, the decoder's outputs determine which channel works and which upper-level signals are chosen as MUX output.

In this CML circuit, there are eight differential inputs in the first level. That caused parasitic capacitance which is somewhat effect the bipolar switching speed. However, comparing to the traditional MUXs with at least three level logic gates delay (shown in Fig 3), this structure takes advantage in operation speed. Furthermore, the new MUX has only one constant current source, while traditional ones have current source for each logic gate. It goes without saying that the power greatly saves.

Due to the bipolar characteristic, even there is only one transistor works at third level, other signal channel also have little current flows. But the leakage current in these channels are very small contrast to the working channel current. When the current source is 350uA, the working channel occupies 344uA, and others have only 6uA in total.

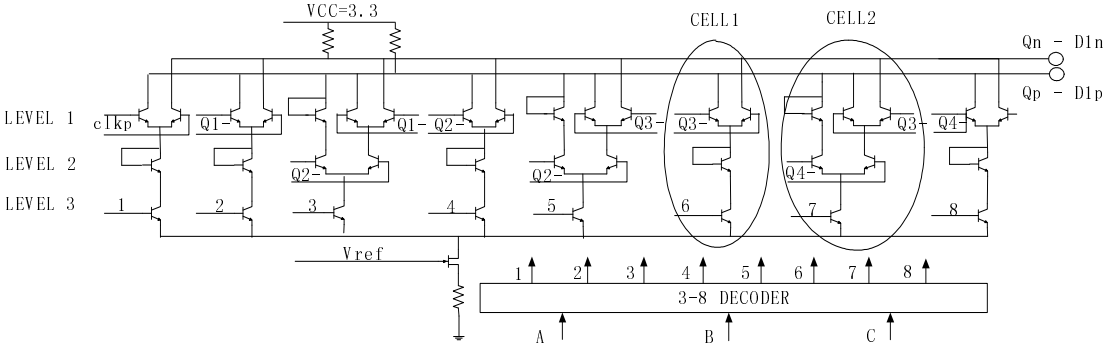

Figure 5-11 "NEW" MUX structure

5.4.3 Programmable Division Ratio

As we discussed earlier in the paper, the leakage current of un-working channel is very small. It could be very easy to expanding the division ratios without function mess.

First, we add more DFFs to the DFF chain. Each DFF we add will increase two more division ratios. Second, expand the MUX circuit. If the division ratio is odd, add CELL1 (Figure 5-11); if the division ratio is even, add CELL2. Each CELL corresponds to one division ratio. The good thing is even adding more cells to the MUX, the total current will keep at 350uA. Consequently, the power dissipation would be still the same. At last, we need to increase the decoder's bits. Three-bit decoder can control eight division ratios, four-bit can control sixteen division ratios and N-bit can control 2^n division ratios.

The expanding operation based on "copy & paste" existing blocks. So it will largely save design and layout time, be good for update quickly.

In some cases, we just need some special division ratio. If the desired ratios are not big, this kind of programmable divider would be a good choice. For instance, divided by /2, /3, /5, /8, are needed, we can use only two control bits to choose these ratios; and only two CELL1 and two CELL2 are enough for building the MUX circuit.

5.4.4 Simulation Result

Simulation of extracted programmable divider is shown in Figure 5-12. The peak to peak sine wave is 200mV, and the input frequency runs at 7.5GHz. When this divider applied in a 5GHz PLL, the maximum input frequency of the divider is limited by the oscillation frequency of VCO. When it comes to the power consumption, the current drawn from the power supply can slightly vary with input frequency and division factor. For 7.5 GHz operation, the divider draws a maximum of 20.8 mA with 3.3 V supply.

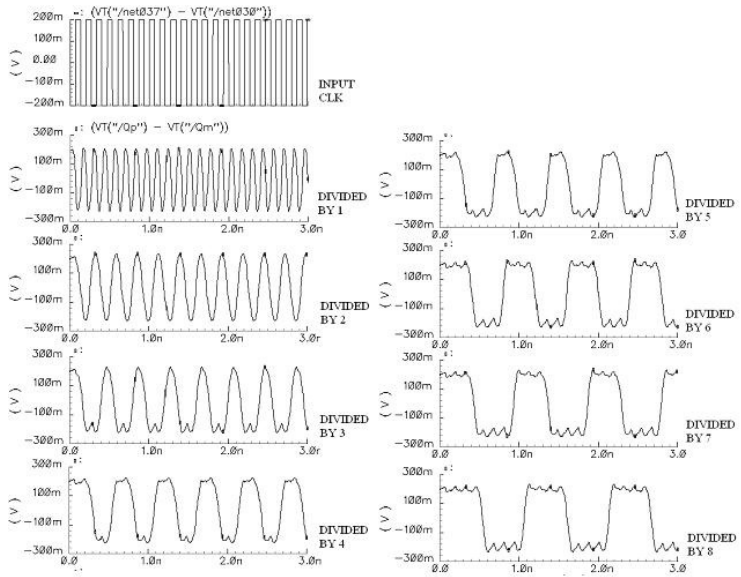

Figure 5-12 Programmable divider simulation

5.5 **Phase Frequency Detector**

A Phase/ Frequency Detector produced an output signal proportional to the phase/ frequency difference of the signals applied to its inputs. It significantly increases the acquisition range and lock speed of PLLs. Sequential phase/frequency detectors (PFDs) generate two outputs that are not complementary. Illustrated in Figure 5-13

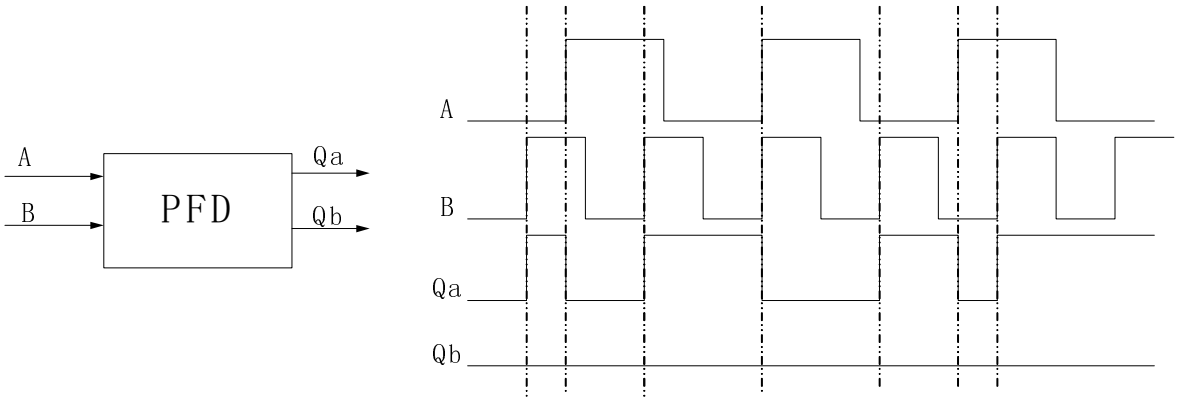

Figure 5-13 Phase/frequency detector response with $W_A<W_B$

The operation of a typical PFD is as follows. If the frequency of input A is greater than that of input B, then the PFD produces positive pulses at Q_A, while Q_B remains at zero. Conversely, if $\omega_A < \omega_B$, then positive pulses appear at Q_B while $Q_A = 0$. If $\omega_A = \omega_B$,

then the circuit generates pulses at either Q_A or Q_B with a width equal to the phase difference between the two inputs. Thus, the average value of Q_A -- Q_B is an indication of the frequency & phase difference between A and B. The outputs Q_A and Q_B are usually called “UP” and “DOWN” signals.

The most common implementation of the Phase/Frequency Detector is shown in Fig5-14 [1]

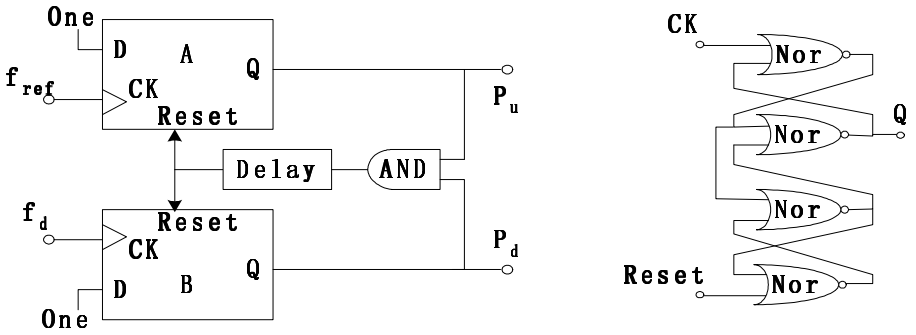

Figure 5-14 Implementation of PFD & DFF used in PFD

The circuit consists of two edge-triggered, resettable D-flip-flops with their D inputs connected to logical ONE. Also it contains AND gate and buffer delay (on the feedback for reducing dead-zone problem) Signals A and B act as clock inputs of DFF_A and DFF_B, respectively. We note that a rising edge of either the reference frequency or the divider output frequency immediately causes the corresponding flip-flop’s output to go high. Once the other output is also high, the AND gate produces a one, which resets both flip-flops to zero states until the next rising edge of either input arrives. We can use NOR gate to build D-flip-flop simply (in Fig5-14).

Simulation of extracted Phase-Detector (Compare 200MHz and 250MHz) is shown in Figure 5-15. The output of a PFD can be converted to DC in different manners. One approach is to use the outputs drive a charge pump.

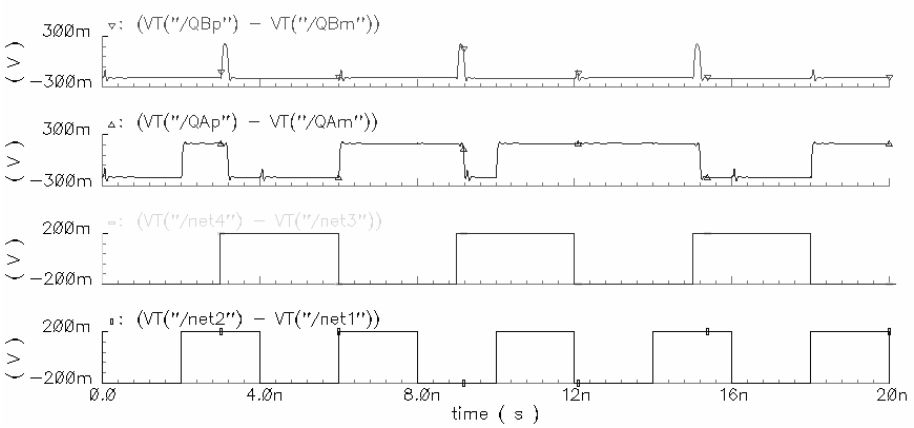

Figure 5-15 Phase/frequency detector simulation

5.6**Charge Pump & Programmable Bias Current Setting**

5.6.1 Charge Pump

A Charge Pump is responsible for placing charge into or taking charge out of the loop filter, and therefore, moving voltage on the VCO up or down.

As the input of charge pump is the output of phase/frequency detector and the PFD use CML, we implement the circuit illustrated in Fig 5-16[2]

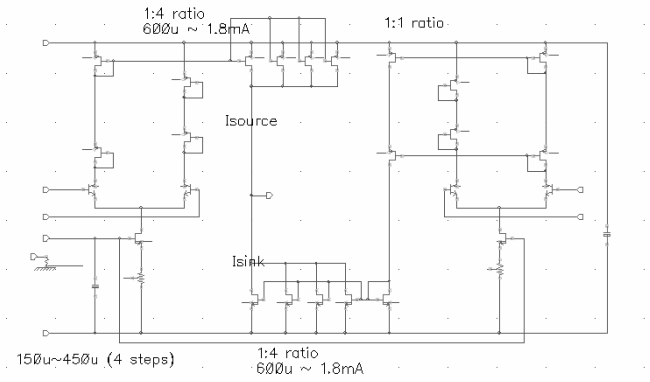

Figure 5-16 Charge pump with cascade current mirrors

With a BiCMOS technology, the input differential pairs are made with bipolar transistor for ease of switching. Others are all CMOS. We use cascade structure here

In this circuit, when both input differential pairs are low, they steer current either to line2 and line 5 and there is no current at the output. Mention that, with the Down signal,

there is an extra current direction reversal because the current must be pulled downwards. When the Up pair is active and the Down pair is zero, the upper CMOS at the output line is turning on, and then the charge pump places charge into loop filter. When the Down one active, the lower CMOS at the output line is turning on, and then the charge pump takes charge from loop filter.

5.6.2 Programmable Bias Current Setting

Often there is a need to be able to adjust the current flowing in the charge pump either because the designer would like the flexibility of being able to adjust the loop bandwidth or because the exact charge pump current for best phase noise performance is not known with certainty before fabrication. The charge pump current can easily be made scalable with a simple circuit such as that shown below. This circuit takes the reference current produced by band-gap and the scale it up. Placing switches in series with the current mirrors allows the current to be programmed in binary steps, such that the output current I is give by : $I = (8p3+4p2+2p1+p0)I_{ref}$ [2].

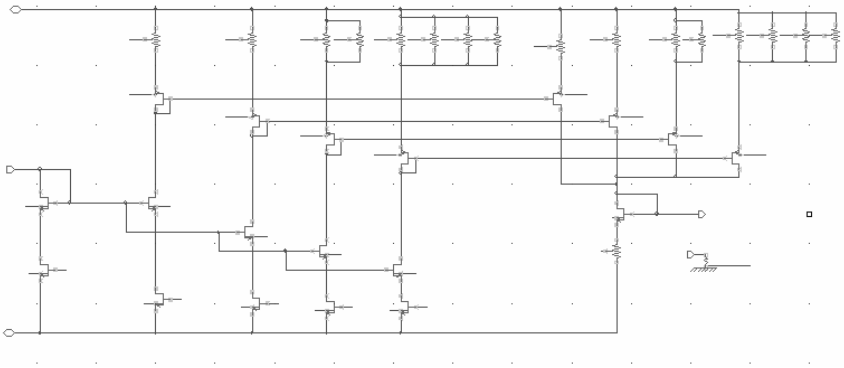

Figure 5-17 Programmable current setting

5.7 **Loop Filter**

As mentioned in Chapter 4, VCO is normally controlled by voltage not current; we need to turn the current produced by charge pump back into a voltage. Furthermore, it is

not desirable to feed pulses into the VCO. Therefore, a low pass filter is needed in the loop, called loop filter.

If using a simple capacitor to convert current to voltage, the loop can not be stable[35]. Normally a combination of capacitors and resistors is used. The typical second order loop filter is shown in Figure 5-18. The frequency response of phase detector, charge pump is mainly determined by the loop filter.

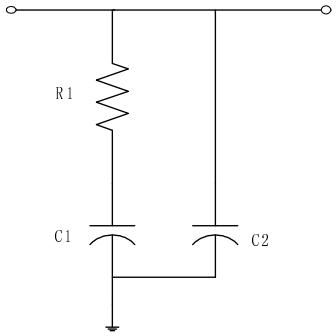

Figure 5-18 A typical second order loop filter

The The transfer function is given by[2]

$$F(s) = \frac{(1 + sRC1)}{s(C1 + C2)(1 + sRC_s)} \quad \text{, where } C_s = \frac{C1C2}{C1 + C2} \qquad (5\text{-}2)$$

There are one zero $\frac{1}{RC1}$ and two poles 0 and $\frac{1}{RC_s}$. Notice that at low frequencies, the response is dominated by the zero in transfer function; thus the circuit acts like an integrator. Frequency response is given in Figure 5-19 and 5-20

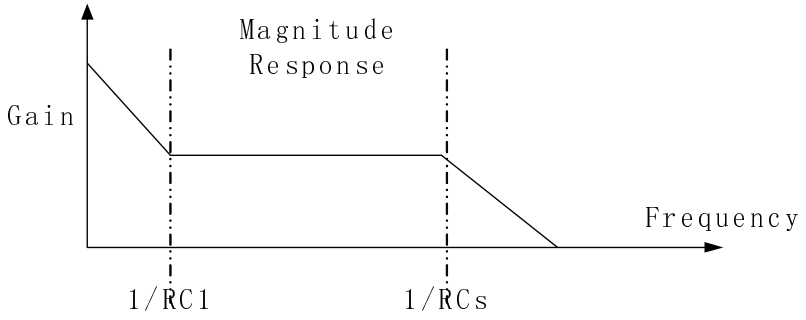

Figure 5-19 Phase detector, charge pump, loop filter magnitude response

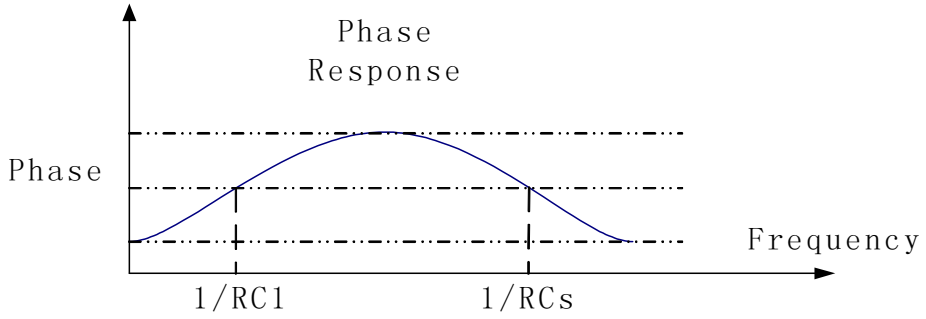

Figure 5-20 Phase detector, charge pump, loop filter phase response

The whole phase detector, charge pump and loop filter simulation is shown in Figure 5-20.

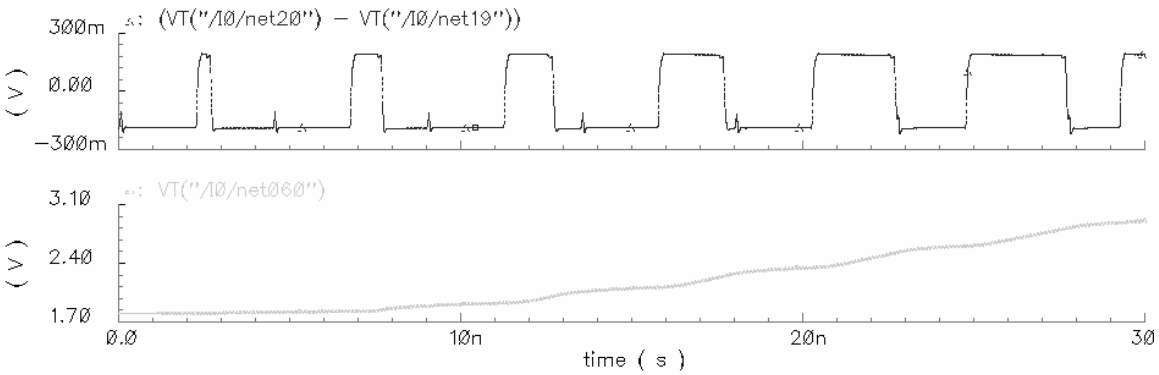

Figure 5-21 Charge pump and phase detector simulation

CHAPTER 6 MEASUREMENTS AND CONCLUSIONS FOR 5 GHZ PLL IN 0.5 UM SIGE TECHNOLOGY

6.1 Background

The role of a frequency synthesizer is to provide a reference frequency for frequency translation in a transceiver. This means that the frequency synthesizer needs to generate a set of frequencies at the frequency bands of the standard. On the other hand, due to the presence of a variety of standards covering different applications and employing a wide range of frequencies, having one portable device working for several standards with low power and low cost is highly desirable. In this research work, we designed a synthesizer with 600MHz ~ 5.2 GHz output frequency. Except the off chip loop filter, a prototype based on the Fractional-N PLL architecture was fabricated in a 0.5μm 1-poly 4-metal SiGe process.

6.2 Design Specifications

The block diagram of the prototype shown in Figure 5-5: Phase Detector, programmable divider and MMD have differential input and differential output. Charge pump has differential input and single output while VCO has single input and differential output. the low-noise buffer following the MMD is differentially clocked by VCO's output.

6.2.1 Frequency Plan

A good frequency plan is crucial to achieving all the specifications of different standards, with a minimal amount of hardware and power consumption. The frequency

plan determines how the frequency translation of the carrier is performed in both receive and transmit paths. Therefore, the frequency plan determines the amount of hardware and power it takes to generate the reference frequency with the frequency synthesizer and has a significant impact on the overall synthesizer performance, namely, phase noise, spurious tones and the required power consumption.

The first design choice is the reference frequency. It is desirable to develop a frequency plan where only one external reference oscillator is used. The phase noise performance of the external oscillator also influences the choice of the reference frequency. Currently, the available crystal oscillators on the market below 200MHz typically have a phase noise level below -145dBc/Hz at 50kHz offset frequency.

The VCO's running range is 4.7GHz~ 5.3GHz and the programmable divider's division ratio is 1~8, thus the output frequency can by used in 802.11a, 802.11b, ETS1/BRAN HiperLAN/2, BlueTooth, CT2/CT2+ and etc[36]..

The MMD's division ratio ranges from 128 to 511. If we use 511 as a division ratio, then the phase noise of the crystal oscillator and phase detector and the divider is amplified by 511. With a big noise enhancement from the divider it is virtually impossible to meet the phase noise requirement for cellular applications using a wideband PLL with an integrated VCO. Furthermore, since the VCO's lowest frequency in this design is 4.7GHz, with a 511 division ratio, the reference frequency would be very small. It would not good for the bandwidth and settling time. Therefore, the MMD ratio is chosen around the 130, and the reference frequency is chosen as 40MHz. The noise amplification of the crystal oscillator, phase detector, and dividers are obviously reduced,

making it possible to implement a wide band PLL using an external low phase noise crystal oscillator.

6.2.2 Loop Parameter Design

The loop bandwidth of the PLL needs to be optimized in order to achieve minimum overall phase noise at the offset frequency where the performance is most critical. A 40MHz reference frequency is chosen for the reason mentioned above. The loop bandwidth of PLL should be less than 1/10 of the reference frequency for the stability of the loop. For this design, the loop bandwidth is chosen to be about *100KHz* for maximum suppression of the noise from the reference, loop filter and phase detector.

Fig. 5.18 shows one way to implement the loop filter based on an RC network. Knowing the desired loop bandwidth, we can determine the RC parameters of the loop filter by leaving enough phase margin for the loop. The value of C_2 is usually less than 1/5 of the C_1 for not increasing the loop settling time. For a 100 KHz bandwidth, it is less than 1/10 of crystal input reference to guarantee the loop stability. Here, we pick R=10K, C_1=11nF, C_2=1.1nF.

6.3 **Layout photo and Simulation results**

The whole chip layout is shown in Figure 6-1. It occupies 3.1×2.56 mm^2 area and consumes 431.97mw under 3.3 V power supply in simulations.

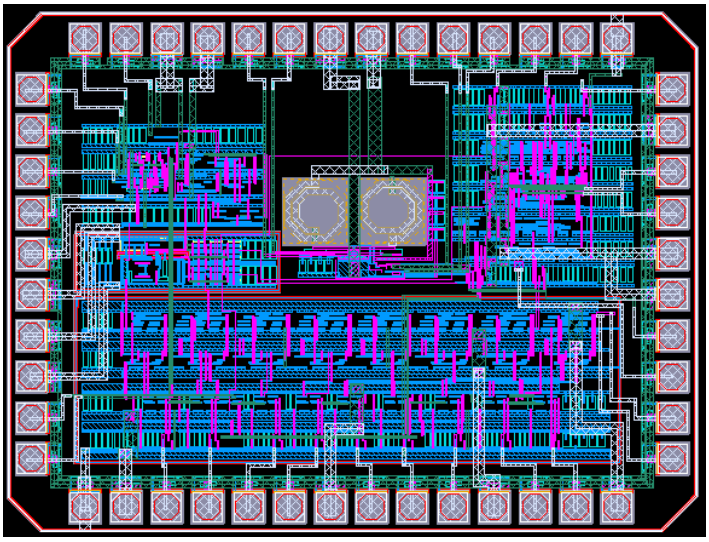

Figure 6-1 Die photo of the PLL in 0.5um SiGe technology

An open-loop simulation results using Cadence are shown in Figure 6-2. The first row shows the output of VCO runs at 4.75GHz with a 3.3V voltage control. The second row shows the output of MMD with a division ratio 256. The peak to peak amplitude is 200mV for a single output of a differential pair. The third row shows a 20 MHz square wave is applied to the reference frequency. The forth row shows the difference between the reference frequency and the MMD outputs at the phase detector outputs. In this case, the reference frequency is much faster than the MMD output, and the VCO has negative gain. Therefore, the voltage at the filter's output should drop to make VCO run faster. The fifth row shows the output of charge pump and loop filter. And the current discharges from the capacitors.

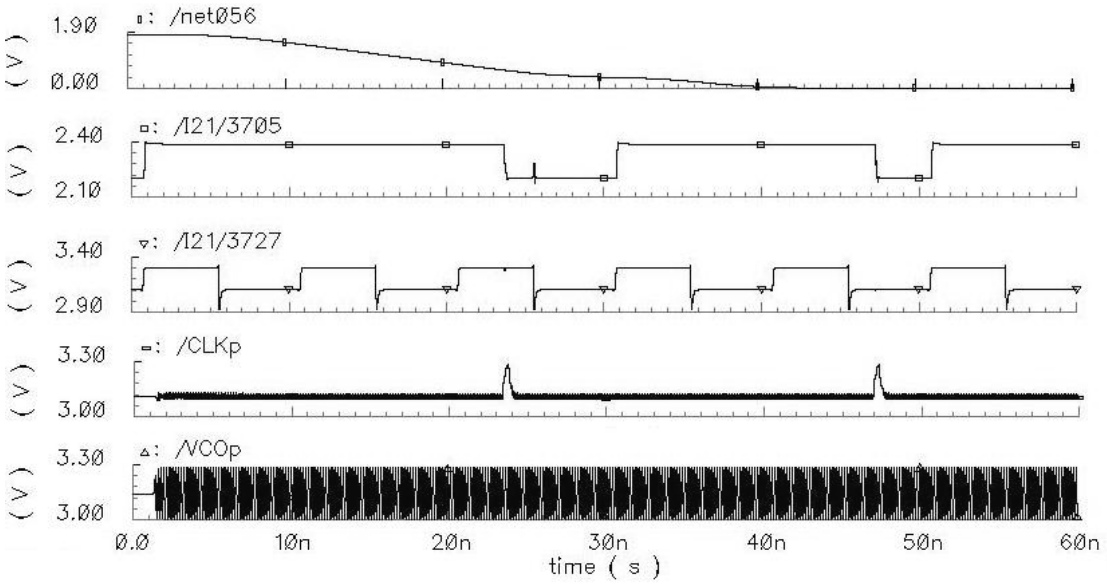

Figure 6-2 PLL open-loop simulation

6.4**Measurements**

6.4.1 Die Photos

The 5GHz PLL was fabricated in a 0.5mm 1-poly 4-metal BiCMOS process with SiGe, shown in Figure 6-3. The Die Size is 3.14×2.62 mm^2.

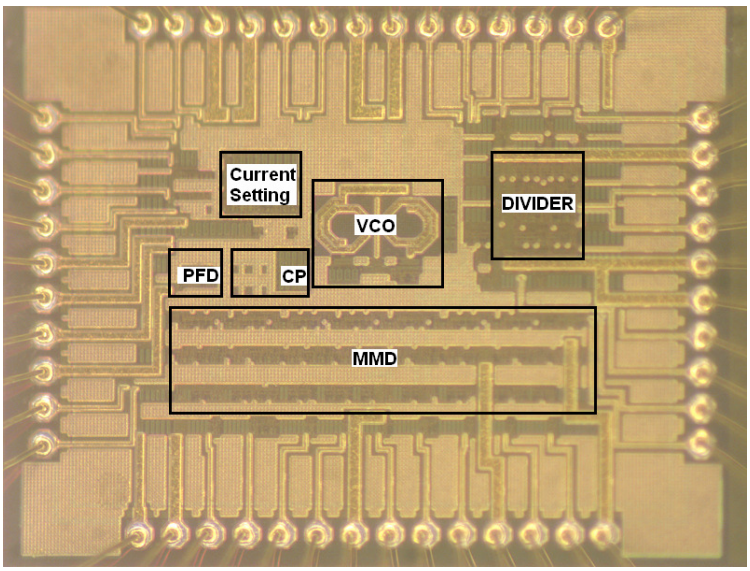

Figure 6-3 Die photo of PLL

And Figure 6-4 shows the PLL with ceramic package photo. This package is a 52 leadless chip carrier package with a 0.300" cavity (topside) and is 0.750" square.

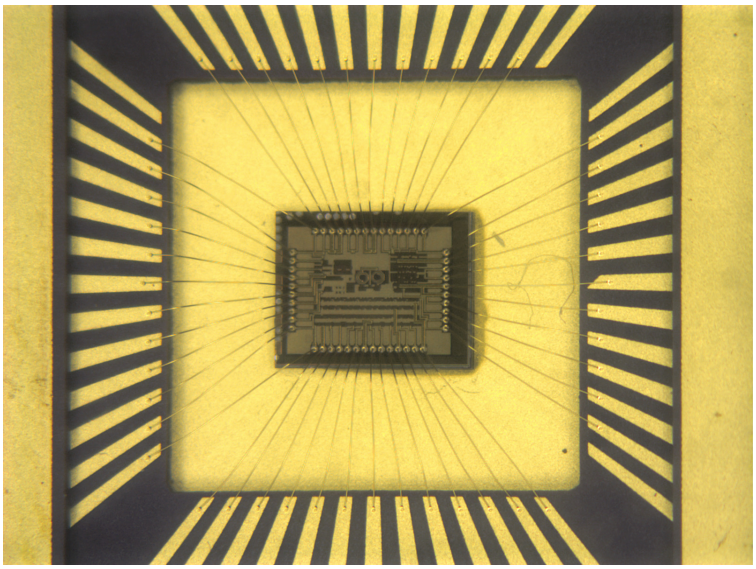

Figure 6-4 PLL with package photo

6.4.2 Printed Circuit Board Design

The PCB is designed for 5GHz PLL using FR4 material with two metal layers. It's about 3 × 3 inches. There are eight power supply pins, seven ground pins and two sub pins. We connect all ground and sub pins together on the PCB; while we separate analog and digital power supplies on the board in order to test different single blocks (one for VCO; one for MMD, one for programmable divider and one for other blocks). For each power supply pin, we leave enough room to place dcaps for reducing high frequency noise.

For VCO outputs, MMD outputs, programmable divider outputs, and crystal reference input, we plan to use SMA connector. In order to decrease the calculation errors, we use 91.5-mil conductor on board to match the 50 Ohm resistance.

For other inputs, outputs or control bits, since they are all DC value, we only need to use metal wire to connect them from board to DC sources.

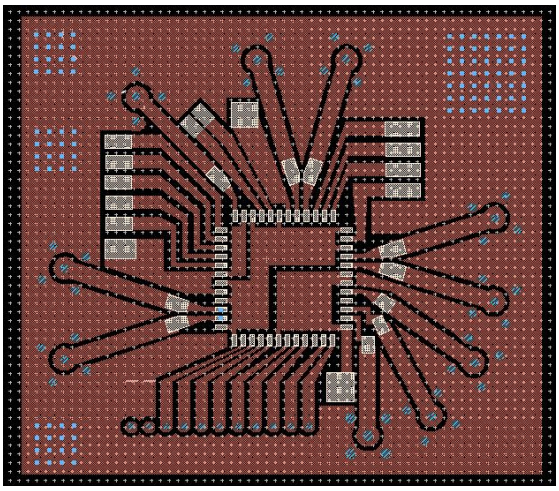

Figure 6-5 First copper layer on PCB

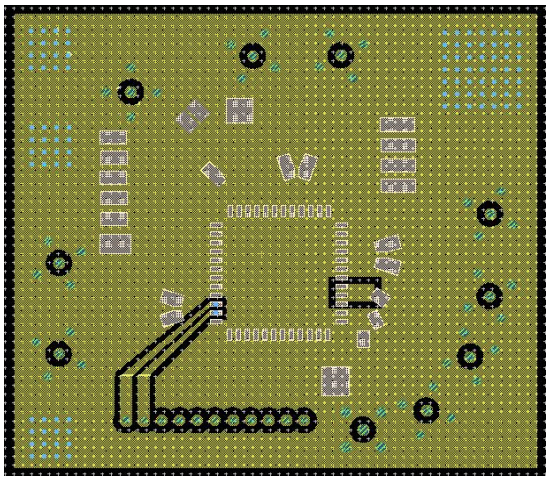

Figure 6-6 Second copper layer on PCB

6.4.3 Results

The digital and analog power supplies all connected to 3.3V. Ground and subtract are tied together. The total current from power supply is 148mA.

Test results showed the VCO's output running at 5.12GHz with 0 V voltage control. And the MMD's division ratio is 256, thus the output frequency of MMD is 20MHz. This output re-clocked by VCO's output to remove noise from the MMD, then feed back to the Phase detector's input. It compared with the reference frequency to determine charging or discharging the filter's capacitor. MMD's spectrum shows in Figure 6-7 and its waveform shows in Figure 6-8. The output signal is about 55dB above the noise floor.

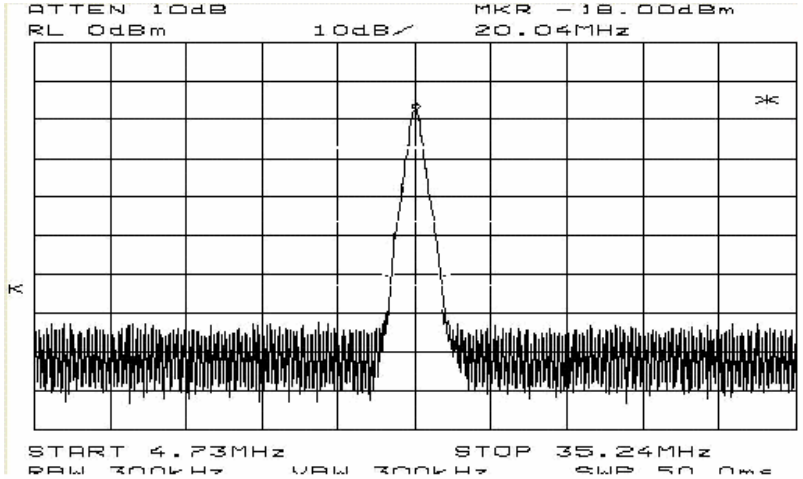

Figure 6-7 Spectrum of MMD's output frequency (division ratio is 256)

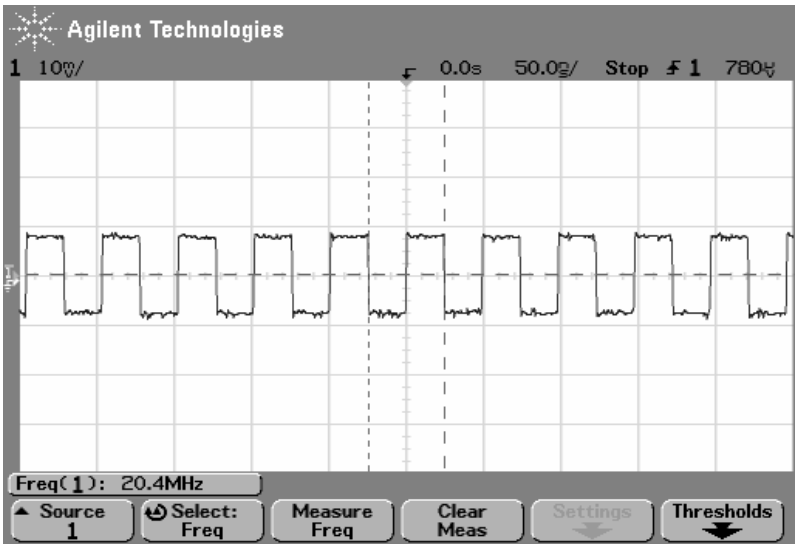

Figure 6-8 Waveform of MMD's output frequency (division ratio is 256)

When the MMD's division ratio changed to 360, the output frequency of MMD is 14.25MHz. Spectrum shows in Figure 6-9 and waveform shows in Figure 6-10. The output signal is about 54dB above the noise floor.

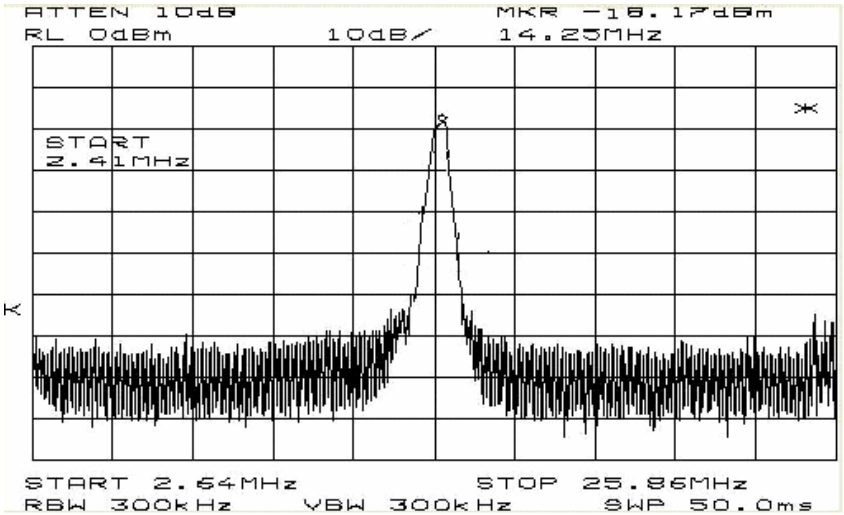

Figure 6-9 Spectrum of MMD's output frequency (division ratio is 360)

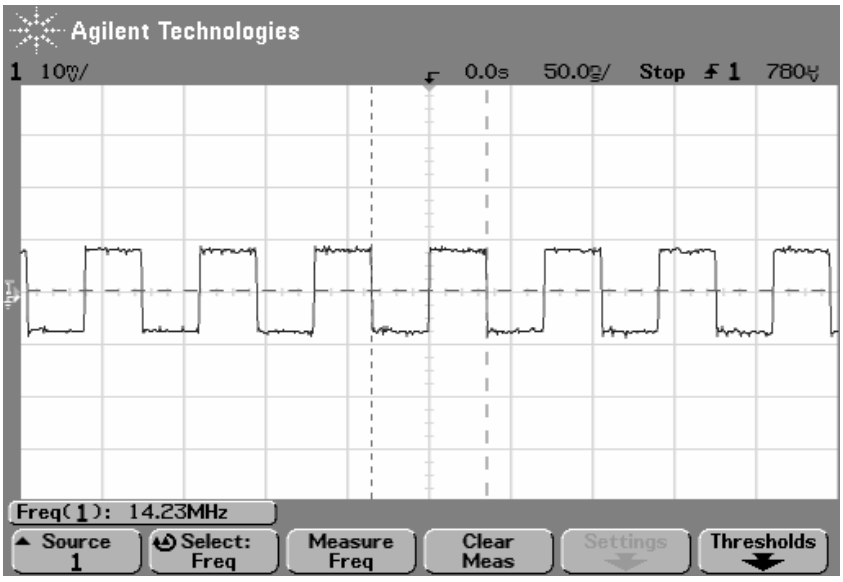

Figure 6-10 Waveform of MMD's output frequency (division ratio is 360)

In order to get wide range of PLL output, we added a programmable divider and a divided by 2 cell following the VCO serially. When the VCO's output equals 5.12GHz and the divider's division ratio is 7, thus the output frequency of MMD is (51200/14)=364.3MHz.. Divider's spectrum shows in Figure 6-11 and waveform shows in Figure 6-12. The output signal is about 60dB above the noise floor.

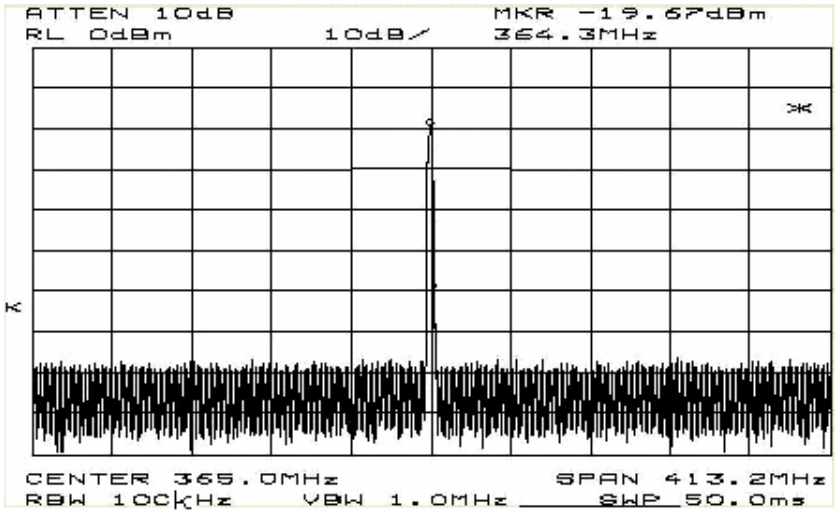

Figure 6-11 Spectrum of programmable divider's output (division ratio is 14)

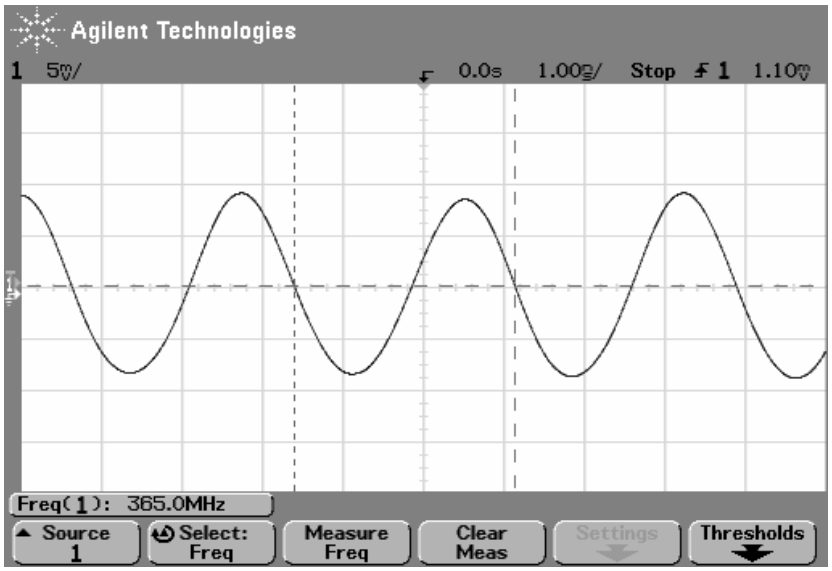

Figure 6-12 Waveform of programmable divider's output (division ratio is 14)

CHAPTER 7 CONCLUSION

In this thesis, two kinds of frequency synthesizers are discussed. And the focus part in this research work is an integer-N PLL based frequency synthesizer. Various circuit techniques to minimize the power consumption and maximum the operation speed are explored.

DDS technique is an easy and flexible way to implement variety digital modulations, which can transmit and receive information with higher accuracy in the presence of noise and distortion. The realization of BPSK, QPSK, OQPSK and MSK are presented. When we used a Gaussian filter before the modulation, the frequency spectrum is obviously shrunk.

Several PLL frequency synthesizers are compared. Each of them has its own advantages and disadvantages. In the integer-N architecture, the loop bandwidth is limited and the input reference must be equal to the channel spacing and the phase noise is worse. While fractional-N PLL has a wide bandwidth, quicker settling time, small channel spacing with a high reference frequency, it suffers fractional spurs. This is often undesirable, so various methods of suppressing the spurs have been devised, such as fractional compensation, randomizing the sequence of the modulus, or using Delta-Sigma modulators.

In the end, RF circuit designs of an integer-N phase locked loop frequency synthesizer in SiGe technology are presented. The building blocks include phase detector,

charge pump, programmable current setting, loop filter, VCO, programmable divider and MMD. Because noise from the VCO is suppressed in wideband PLL architectures, other noise sources become more important in the synthesizer performance. Noise from the crystal oscillator reference and phase detectors become the most important contributors within the loop bandwidth and are referred to the output enhanced in effect by the divider ratio N.

In this PLL design, all digital blocks are used CML structure, which can run over 10GHz. Except the loop filter, the whole loop is integrated in a 0.5um BiCMOS technology. With a regular 3.3V power supply, the whole chip consumes 476.3mW. The VCO's operation range is 4.7GHz~5.3GHz; programmable divider's maximum speed is 7.5GHz; MMD's maximum speed is 8.3GHz, and phase detector can reach 500MHz.

BIBLIOGRAPHY

[1] Behzad Razavi, *RF Microelectronics*. 1998 Prentice-Hall PTR.

[2] John Rogers, Calvin Plett, and Foster Dai, *Integrated Circuit Design for High-Speed Frequency Synthesis*, Boston, MA: ARTECH HOUSE PUBLISHERS, Feb. 2006.

[3] Roland E. Best, *Phase-Locked Loops: Design, Simulation, and Applications,* 2003 The McGraw-Hill Companies, Inc

[4] A. A. Abidi, "RF CMOS Comes of Age," *2003 Symposium on VLSI Circuits Digest of Technical Papers*.

[5] Behzad Razavi, "CMOS Technology Characterization for Analog and RF Design," *IEEE J. Solid-State Circuits*, VOL. 34, NO.3, pp.268-276, Mar. 1999.

[6] Peter Ashburn, *SiGe Heterojunction Bipolar Transistors* ,2003 John Wiley & Sons Ltd, England

[7] D. J. Friedman ET AL, "SiGe BiCMOS integrated circuits for high speed serial communication link," *IBM J. Res. & Dev.*, Vol. 47, No. 2/3, pp.259-282, Mar. 2003

[8] Bar-Giora Goldberg, *Digital Frequency Synthesis Demystified : DDS and Fractional-N PLLs,* 1999 by LLH Technology Publishing

[9] Cheng, S.; Jensen, J.R.; Wallis, R.E.; Weaver, G.L.;" Further enhancements to the analysis of spectral purity in the application of practical direct digital synthesis", *Frequency Control Symposium and Exposition, 2004. Proceedings of the 2004 IEEE International,* 23-27 Aug. 2004 Page(s):462 – 470

[10] Henry T. Nicholas, III and Henry Samueli, "An Analysis of the Output Spectrum of Direct Digital Frequency Synthesizers in the Presence of Phase-Accumulator Truncation," *41st Annual Frequency Control Symposium*, pp.495-502, 1987.

[11] J. Jiang and E. Lee, "A Low-Power Segmented Nonlinear DAC-Based Direct Digital Frequency synthesizer," *IEEE J. of Solid-State Circuits*, vol. 37, Oct 2002.

[12] Siamak Mortezapour and Edward K.F.Lee, "Design of low-power ROM-less direct digital frequency synthesizer using nonlinear digital-to-analog converter," *IEEE J. Solid-State Circuits*, vol. 34, pp. 1350–1359, Oct. 1999.

[13] Stephen G. Wilson, *Digital Modulation and Coding,* 1996 Prentice-Hall, Inc

[14] Lawrence E Larson "RF and Microwave circuit design for wireless

[15] Wai-Kai Chen, *The Circuits and Filters Handbook, Second Edition,* 2002 CRC Press LLC

[16] Nicholas John Loy, *An Engineer's Guide to Fir Digital Filters,* Prentice Hall; 1st edition (July 14, 1997)

[17] F.M. Gardner, *Phaselock Techniques*, 2nd ed., New York: John Wiley, 1979.

[18] F. M. Gardner, "Charge-Pump Phase-Locked Loops," *IEEE Trans. Comm.*, Vol. COM-28, pp.1849-1858, November 1980.

[19] S. Pamarti, "Techniques for Wideband Fractional-N Phase-Locked Loops," PhD Dissertation, University of California, San Diego, 2003.

[20] L. Lin, L. Tee, P. R. Gray, "A 1.4 GHz differential low-noise CMOS frequency synthesizer using a wideband PLL architecture,", *IEEE ISSCC Digest of Technical Papers*, pp. 204-205, Feb. 2000.

[21] Philips Semiconductor, *Philips RF/Wireless Communications Data*

[22] Riley, T.A.D.; Copeland, M.A.; Kwasniewski, T.A.; "Delta-sigma modulation in fractional-N frequency synthesis" *Solid-State Circuits, IEEE Journal of* Volume 28, Issue 5, May 1993 Page(s):553 – 559

[23] Woogeun Rhee, Bang-Sup Song and Akbar Ali, "A 1.1-GHz CMOS Fractional-N Frequency Synthesizer with a 3-b Third-Order ΔΣ modulator," *IEEE Journal of Solid-State Circuits*, Vol. 35, No.10, 2000.

[24] Alioto, M.; Palumbo, G.; "CML and ECL: optimized design and comparison" *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on [see also Circuits and Systems I: Regular Papers, IEEE Transactions on]*, Volume 46, Issue 11, Nov. 1999 Page(s):1330 – 1341

[25] Payam Heydari, Ravindran Mohanavelu, "Design of Ultrahigh-Speed Low-Voltage CMOS CML Buffers and Latches," *IEEE Transactions on Very Large Scale Integration System*, vol. 12, No. 10, pp. 1081- 1093, Otc. 2004.

[26] Harada, Y. "Delay components of a current mode logic circuit and their current dependency" *Solid-State Circuits, IEEE* Journal of Volume 30, Issue 1, Jan 1995 Page(s):54 – 60

[27] Paul R. Gray, Paul J. Hurst, Stephen H. Lewis and Robert G. Meyer, *Analysis and Design of Analog Integrated Circuits*. John Wiley & Sons, INC.2001.

[28] Richard C. Jaeger, *Microelectronic Circuit Design*. The McGraw-Hill Companies, Inc., 1997.

[29] Cicero S. Vaucher, Igor Ferencic, Matthias Locher,Sebastian Sedvallson, Urs Voegeli, Zhenhua Wang, "A Family of Low-Power Truly Modular Programmable Dividers in Standard 0.35-µm CMOS Technology," *IEEE Journal of Solid-State Circuits*, vol. 35, No.7, pp.1039-1045, July 2000.

[30] C. S. Vaucher and D. Kasperkovitz, "A wide-band tuning system for fully integrated satellite receivers," *IEEE J. Solid-State Circuits*, vol. 33, pp. 987–998, July 1998.

[31] H. Sheng *et al.*, "A high-speed multimodulus HBT prescaler for frequency synthesizer applications," *IEEE J. Solid-State Circuits*, vol. 26, pp. 1362–1367, Oct. 1991.

[32] C. S. Vaucher and Z. Wang, "A low-power truly modular 1.8 GHz programmable divider in standard CMOS technology," in *Proc. 25th Eur. Solid-State Circuits Conf.*, Sept. 1999, pp. 406–409.

[33] Ingo Schafer, "Synthesis of Multilevel Multiplexer Circuits for Incompletely Specified Multioutput Boolean Functions with Mapping to Multiplexer Based FPGA's" *IEEE Transactions on computer-aided design of integrated circuits and systems,* vol. 12, No 11, November 1993

[34] Y. Deng Z., Zhou B. Deng; *Digital Circuit Design Hand Book*, 1998, pp.45-46.

[35] Dan H. Wolaver, *Phase-Locked Loop Circuit Design,* 1991 PTR Prentice Hall

[36] N. Filiol, et, al., "A 22 mW Bluetooth RF transceiver with direct RF modulation and on-chip IF filtering," Digest of Technical Papers, *IEEE International Solid-State Circuits Conference*, vol. 43, pp. 202-203, Feb. 2000.