UNIVERSITY OF NAIROBI

DEPARTMENT OF ELECTRICAL AND INFORMATION ENGINEERING

FINAL YEAR PROJECT
PROJECT NO. 085

TITLE:
A PHASE-LOCKED LOOP FREQUENCY SYNTHESIZER

BY:
TUNDULI W MICHAEL
REG. NO.: F17/2143/04

SUPERVISOR: Dr. V. K ODUOL
EXAMINER: Mr. C.OMBURA

This is a final year project report submitted in partial fulfillment for the award of the degree of Bachelor of Science in Electrical and Electronics Engineering.

May 2009
DEDICATION

This project is dedicated to all the people in my life who have made it possible for me to reach this far, especially my family, friend and teachers.
ACKNOWLEDGEMENTS

My gratitude is to my family for the support, encouragement and assistance without which I may never have come this far, especially Arnold and Felician.

My no lesser gratitude is to teachers and friends who see a potential in me and encourage me to keep on, my supervisor Dr. V. Oduol for directing and keeping me on the way when doing this project.

Last and most important is to GOD, for life, talents and abilities He has given me.
3.4 Phase Locked Loop Frequency Synthesizer é é é é é é é é é é é é é é  .16
   3.4.1 Phase detector and charge pumpé é é é é é é é é é é é é é é ..17
   3.4.2 Crystal oscillator and VCOsé é é é é é é é é é é é é é é ....é  22
   3.4.3 Loop filtersé é é é é é é é é é é é é é é .. é é é é ..22
   3.4.4 Dividersé é é é é é é é é é é é é é é .é é é é é é .. é ....24
   3.5 Key Parametersé é é é é é é é é é é é é é é .é é é é é é .24

CHAPTER FOUR
INTEGER-N AND FRACTIONAL-N SYNTHESIZERS....................... 26
   4.1 Introductioné é é é é é é é é é é é é é é é é é é .... éé 26
   4.1.1 Integer-N Frequency Synthesizeré é é é é é é é é .... éé 27
   4.1.2 Shortcomings of Integer-N Synthesizeré é é é é é é ..é é ..31
   4.2 Fractional-N Synthesizeré é é é é é é é é é é é é ..é é ..31
       4.2.1 Introductioné é é é é é é é é é é é é ..é é ......31
       4.2.2 Types of Fractional-N synthesizeré é é é é é é ..é é ..32
       4.2.3 Advantages of Fractional-N synthesizersé é é ....é é ..é .37
       4.2.4 Summaryé é é é é é é é é é é é é é é ....é é .....38

CHAPTER FIVE
DESIGN AND ANALYSIS......................................................... 39
   5.1 Design problemé é é é é é é é é é é é é é é é é é é ..39
   5.2 Design Valuesé é é é é é é é é é é é é é é é é é ..é 39
   5.3 Design circuité é é é é é é é é é é é é é é é é é ..é é é é .41
   5.4 Results and Displays

CHAPTER SIX
CONCLUSION AND RECOMMENDATIONS............................... 48
REFERENCES...........................................................................49
ABSTRACT
This report looks into phase locked loop frequency synthesizers. It introduces phase locked loop frequency synthesizers in chapter one superficially, a short history of its development and applications. Chapter two deals with frequency synthesis, chapter three phase locked loop (PLL) frequency synthesizer (FS) at a deeper level while chapter four deals with the two major types of PLL synthesizers: integer-N and fractional-N. The design problem is dealt with in chapter five and chapter six concludes as well as puts forth a few recommendations.
Frequency synthesis is an exciting and dynamic field, with evolving challenges. In as much as much attention is now given to direct digital synthesis (DDS) still PLL synthesis will reign until DDS advantages outweighs PLL's.
CHAPTER ONE

INTRODUCTION

1.1 INTRODUCTION

Phase locked loop (PLL) frequency synthesis is the most commonly used method of producing high frequency oscillations in modern communications equipments.[1] This is done through the use of a PLL frequency synthesizer which comprises of basically:

a) A Voltage Controlled Oscillator (VCO)
b) A Phase Detector (PD)
c) A Divider
d) Loop Filter (LP)

It is as shown in figure 1.1

Figure 1.1 Basic PLL frequency synthesizers

It is a control system whereby the output and an input are compared and a correction made. This finds application in many communications equipments where it is required to faithfully produce a certain frequency. If the input reference signal is from a quartz crystal oscillator the output signal is very stable since crystal oscillator is very stable producers of frequencies. With the PLL
1.2 HISTORY

The first PLLs

Basic PLL has been known and widely utilized since it was first proposed in 1922. [4]A phase-locked loop (PLL) frequency synthesis has been with us a long time. In 1932 British radio engineers developed a new type of receiver to challenge the then dominant and still super heterodyne receiver, so they decided to call it homodyne or synchrodyne. The idea was simply a receiver made up of a mixer and a local oscillator followed immediately by an audio amplifier. [1]

In 1932 when a French engineer de Bellescize presented it as a paper in the Onde Electrique. [2, 3]But unfortunately due to its cost of implementation, it didn't become popular until 1960s with the advent of integrated circuits.

Problems occurred in trying to keep the local oscillator on the same phase and frequency as the input signal. To keep in check the local oscillator drift the output of the local oscillator was fed together with a sample of the input signal to a phase detector. The output of the phase detector was a correction voltage applied to the local oscillator to keep on frequency. [1] This type of circuit is that which led to the evolution of PLL. The homodyne receiver was superior to the super heterodyne but the cost of the PLL circuit was not worthy the advantages so the idea was shelved.

Developing the PLL
It stated with the discovery of the effect of electric field on certain crystals by Pierre Curie and his brother Jacque. The Piezoelectric Effect as is called is where when pressure is applied to certain crystals electric field charges are produced and conversely when electric field is applied there is a distortion of in the crystal structure.

With the invention of wireless communication there was a need to have accurate frequency determining device. So the crystal oscillator evolved with its excellent frequency stability.

But it had one major limitation for being used for a narrow range of frequencies. The crystal frequency could only be varied slightly by use of a variable capacitor or circuits designed to respond to the harmonics of the crystal’s fundamental frequency. This method doesn’t work for large range of frequencies. Hence for large range of frequencies and maintain frequency stability two crystals could have their outputs mixed to produce two new frequencies-i.e. sum and the difference of the two original frequencies.

But for many channels this method becomes expensive due to many crystals used.

**Voltage Controlled Oscillator**

There are oscillators that operate over a large range of frequencies. Variable frequency oscillators (VFOs) are made to change frequency by changing the value of one of the frequency determining circuits. A voltage controlled oscillators (VCOs) are ones which the frequency determining component is made to change electrically by an applied voltage. But the problem is that the VCO is not stable, but if we could find a way to combine the VCOs flexibility and the stability of the crystal oscillator then we would have an ideal frequency synthesis system.

But suppose we fed the output of a VCO and a crystal oscillator into a phase detector? If the two signals are equal in phase and frequency, there should be no output from the phase detector. On the other hand if the signals are not in phase, the difference is converted to a DC output voltage. The greater the frequency/phase difference between the two signals, the larger the output voltage. Now suppose this output voltage is coupled to back to the VCO in such manner that it drives the output of the VCO towards the crystal oscillator frequency-eventually the VCO will
lock onto the crystal oscillator frequency. This is a PLL in its most basic form and is shown in figure 1.2

But once the VCO is locked onto the crystal oscillator, it behaves as if it were a fixed frequency oscillator, giving as the stability of the crystal oscillator but losing its flexibility that is desired.

Now suppose our crystal is 12Mhz, but we want the VCO to operate on 24Mhz. The phase detector of course will note the difference and the VCO will pull it to 12Mhz, but what if we could fool the phase detector into thinking the VCO was real operating at 12Mhz when it real is on 24Mhz. If we divide the output fed to the Phase detector by two, the VCO will be forced to operate at 24Mhz so that after division the inputs to the PD are same. This is a PLL frequency synthesizer in its basic form. Figure 1.2a shows this.
Figure 1.2a Basic PLL frequency synthesizer

It is worthy mentioning how PLL has developed to its present form. The first PLL IC was developed around 1965, and it comprised of a four quadrant multiplier as a phase detector, loop filter from passive or active filter and a VCO, is referred to as linear PLL. Around 1970, the four quadrant multiplier was replaced by a digital phase detector which could be an EXOR or JK flip-flop and is referred to as a digital PLL. A few years later, a PLL IC was built from digital blocks and no passive components like resistors and capacitors and is referred to as an all digital PLL. Now another type of PLL is possible, software PLL. Here the function of the specialized hardware of a PLL circuit is performed an algorithm run on a computer. Software PLL requires a microcontroller and a digital signal processor. Its major advantage is flexibility, as through algorithms it is possible to effect any PLL you want. [3] Fractional-N synthesizer IC was first developed in 1994.

1.3 CLASSIFICATION

There are different ways of classifying PLL synthesizers and they include:

a) Method of making them

b) The type of the divider
c) The range of frequency of operation

Under method of making we have

1) Linear PLL synthesizers (L PLL)
2) Digital PLL (D PLL)
3) All Digital PLL (AD PL)
4) Software PLL

Under the type of divider we have

1) Integer-N PLL
2) Fractional-N PLL

1.4 APPLICATIONS OF PLL SYNTHESIZERS

Due to low cost, low power consumption and stability in frequency generation PLL has became a common device in modern communication systems especially in the wireless arena. It finds application in the following places:

1) Satellites
2) Radar systems.
3) Mobile phones
4) Radio links
5) Wireless LAN
6) Home entertainment systems

7) Car radios

In all wireless communication systems frequency synthesizers are needed.
CHAPTER TWO

FREQUENCY SYNTHESIS

2.1 INTRODUCTION

Frequency synthesis is the engineering discipline dealing with the generation of multiple signal frequencies, all derived from a common reference or time base. [5] Generally sine waves are applied in all radio applications, communications, electronic imaging and more. There are two ways of generating a wave form: building the wave form ‘ground up’ digitally (generating all its parameters such as phase, frequency and amplitude (direct synthesis) or use an existing signal to generate another (indirect synthesis).

Frequency synthesis is a mature discipline with lots of texts on it and you may be wondering why still study it? It is still studied because of changing technology and the usefulness of accurate generation of waveforms playing important role in electronic equipment, radar and home entertainment.

Generation and modulation of sine wave is also very important due to the widespread applications in radio, communications, radar, digital communications, electronic imaging and more. [6]

2.2 METHODS OF FREQUENCY SYNTHESIS

Three methods are being currently used for frequency synthesizers (FS). They are:

1) Phase Locked Loop (PLL) synthesis

2) Direct Analogue (DA) Frequency synthesis

3) Direct Digital Synthesis (DSS)

2.2.1 Phase Locked Loop

This method will be dealt with more fully in the next chapter. But it suffices to say here that it is the most popular method. It finds application most sophisticated radar systems or demanding
satellite communication terminals as well as car radios and stereo systems for home entertainment. It is basically a feedback system locking its output frequency to a reference frequency. Its popularity is due to its simplicity and low price. [6]

2.2.2 Direct Analogue (DA) Frequency Synthesis

Here a group of frequency is derived from the main frequency, and these frequencies are mixed and filtered, added, subtracted, or divided according to the required output. We have no feedback mechanism in this method. This method has an advantage of offering excellent spectral purity, especially close to the carrier and excellent switching speeds a critical parameter in many designs determining how fast the synthesizer can hop from one frequency to another.

But DA method is more complicated than PLL to execute therefore more expensive. It finds applications in medical imaging and spectrometers, fast switching antigay communications and radar, electronic warfare (EW) simulation, automatic test equipment (ATE), radar cross-section (RCS) measurement, and other places where the advantages of DA technique are a must at a premium cost. [6]

2.2.3 Direct Digital Synthesis (DDS)

This is a digital signal processing (DSP) discipline. In this method digital circuitry and techniques are used to create, manipulate and modulate a signal digitally and eventually convert the digital signal to its analogue form using a digital analogue converter (DAC). Though the direct digital synthesizer was invented almost 40 years ago, it started to attract attention in the last 20 years because of the enormous evolution of digital technology that started then. The techniques and tools in this area evolved, and it has developed into economical, high performance tool and this now a major frequency synthesis method used in almost all synthesizer designers from instrument makers to applications like satellite communications, radar, medical imaging, and cellular telephony and amateur radio.
It offers fast switching speed, high resolution (the step-size of the synthesizer), small size and low power, good economics, and reliability and producibilty of digital designs. Also since the signal is manipulated digitally it is easy to manipulate and achieve accuracies not attained by analogue techniques and to conventionally interface with the computing machines that usually control the synthesizer.

A major limitation in this method has been inability to extend the range of synthesized frequencies into gigahertz; hopefully this will be achieved in the future. Fractional-N PLL resembles DDS in almost all aspects and operates as a DDS inside the PLL architecture. Figure 2.2.3 is a block diagram of a direct digital synthesis.

![Block diagram of direct digital synthesis](image)

Figure 2.2.3 block diagram of direct digital synthesis
2.3 SYNTHESIZER PARAMETERS

Like any other engineering products, a frequency synthesizer (FS) needs to meet a set of certain specifications. The importance of specifications will depend on which area the synthesizer is applied e.g. for a car radio it should be moderately accurate, extremely reliable, very small and inexpensive, while a FS used in magnetic resonance imaging (MRI) must be very accurate, must have a very high spectral purity, must be able to hop from frequency to frequency very quickly and needs different modulation capabilities. Also the environment where the equipment is to be used has to be taken into consideration. All this makes it a need for designers to compare specifications to the best economical and practical solution.

General specifications include:

1) **Frequency range.**

It specifies the output frequency range including the lower and higher frequency that can be obtained from FS. Units for frequency are hertz or cycles per second.

2) **Frequency Resolution.**

It is also referred to as the step size it specifies the minimum step-size of the frequency increments.

3) **Phase Noise.**

Every signal we generate is derived usually from an oscillator that is usually positive feedback amplifiers with a resonance circuit in their feedback path. Since noise always exists in the circuit, upon power up, this noise is amplified in the resonator band until a level of saturation is achieved. Thus the quality of the signal is usually determined by the resonator Q.

This parameter specifies the phase of the output carrier relative to an ideal output. Phase noise is a major parameter and can be expressed in different ways: measuring FM noise given in hertz, root mean square (RMS) or phase noise in degrees RMS, measure integrated noise in a given bandwidth around the carrier but excluding ± Hz around the carrier, and another method is to measure in the time domain and is referred to as the Allan Variance; by measuring the time fluctuations it is possible to infer the spectrum of the signal.
4) **Switching Speed.**

Specifies the speed at which the FS can hop from frequency to frequency. We have many different definitions for this parameter, they include a time it takes to settle within a specified frequency (± x Hz) from the desired new frequency, time it takes the output phase to settle to 0.1 radians of the final phase.

5) **Phase Transient.**

It is not defined in most applications, however many applications need to define the transient characteristics carefully. Two typical requirements are I) Phase-continuous switching used in measurements, EW, radar and specific modulation (e.g. minimum shift keying). Phase-continuous switching is smooth and generates little noise a factor desirable in systems and network. II) Phase memory switching, it means if FS is running at f1, and then switched to f, f3, f4, etc and back to f1, then it will resume the phase where it would have been if it were running continuously at f1. Applications that require this feature include coherent pulse Doppler radar imagers that require frequency hop but use coherent pulse detection (for predetection integration).

6) **Harmonics**

This parameter specifies the level of the harmonics of the output frequency and depends on many components inside the FS. It is expressed in dB relative to the output frequency (carrier) output power.

7) **Spurious output.**

It defines the level of any discrete output frequency spectral line not related to the carrier. Harmonics are mostly not considered as spurious, but sub harmonics because of multiplications are considered spurious. This parameter is expressed in decibels relative to the carrier output power. Unlike noise, spurious signals only discrete spectral lines not related to the carrier, meaning that they exhibit periodicity.

8) **Control and Interface.**
It specifies control methodology and the interface to the FS. The control can be binary-coded decimal (BCD) or binary; it can be parallel or via a bus (usually 8-bit bus) or serial; it can be transparent or latched. When latched, there is a register that receives the control word and upon activation (by a latch command) loads the control word into the FS also referred to as double buffering.

Some FSs use positive logic and others use negative; and in many general-purpose instrument. GPIB or IEEE-488 is currently the standard interface. VXI is the emerging new standard for instrumentation.

Most single-chip synthesizer, especially PLL make extensive use of the serial interface to allow small packages and highly integrated functionality.

9) Output level

The output power level is usually expressed in decibel. The output power can be either fixed, say, +10dBm, or can cover a range, say -120 to +15dBm. This specification will also include the output power resolution, for example, 1dB or 0.1dB.

10) Output flatness.

It specifies the flatness of the output power and is measured in decibels. For instance; the output power is specified 10dB ±1dB, where dB means decibels over one milliwatt (mW).

11) Output impedance.

This parameter specifies the nominal output impedance of the FS and usually also the recommended load impedance. In most radio-frequency and microwave equipment, this is 50 ohms ($\Omega$). In video it is usually $75\Omega$ and in audio equipment $600\Omega$. 
12) **Standard reference.**

All synthesizers use a reference time based input, this specifies the reference frequency (usually 5 or 10MHz, but there are many others) and its parameters such as stability, phase noise, spurious signals and power levels.

13) **Auxiliary specifications.**

These relate to the specific synthesizers, they include parameters such as size, power supply requirements, environmental factors, quality and reliability.

The most important parameters are:

1) Phase noise

2) Spurious signal level

3) Frequency range and step-size

4) Switching time

5) Size, cost and power dissipation

### 2.4 COMPARATIVE ANALYSIS

It is worthwhile to look at the different advantages and disadvantages of synthesis techniques mentioned in section 2.2.

The PLL techniques is very wideband, it operates from audio frequencies up to millimetre waves and depends on the frequency of the oscillators. PLL synthesizers are relatively simple; moderate to good switching speeds can be achieved; and they are low-cost and easy to apply in most analogue modulations. Single-chip PLL provide a high level integration and low cost. However, resolution is complex to achieve especially with the traditional PLLs, good quality oscillators are quite bulky, and digital modulation is complicated to apply with sufficient accuracy.
DA techniques are very wideband; via multiplications the signal can be generated up to about
100 GHz, and it will probably go further as microwave and millimetre wave technology evolves.
Very high switching speeds are achieved, and the spectral purity is excellent, especially close to
the carrier. However DA synthesis is quite bulky, requires much hardware, and is expensive
(sometimes very expensive) and digital or analog modulation are complicated to apply.

Direct Digital Synthesizers have limited bandwidth, approximately 400 MHz They are simple
and compact, and resolution comes almost free. They have a very high switching speed, phase-
continuous switching and digital producibility. However the bandwidth is still limited but
improving and spurious response is affected by quantization and DAC performance.

It is interesting to note that the three techniques complement one another and this is the reason
designers are combining them more and more, mainly hybrids of PLL and DDS, to achieve wide
bandwidth and good resolution. DA and DDS synthesis are integrated to achieve the speed,
resolution and capability of digital modulation.

Both PLL and DA synthesis have achieved a high level of maturity while DDS is emerging
technology still. In PLL synthesis there has been a major improvement with the introduction of
the fractional-N technique which has great similarities to DDS where it is a part of the phase
locked circuit. DDS has emerged from limited use novelty into a major technology and popular
synthesis technique. This has been partly due to the improvement in the digital technology, the
introduction of low integrated circuits and the evolution of data conversion devices, especially
DAC technology. Introduction of high-speed, high-performance DAC and DDS into a single
low-cost CMOS chip enables this technology to reach a much wider range of applications.

2.5 CONCLUSION

As the requirements of communication systems, radar, medical and industrial equipment become
more stringent, frequency synthesis must evolve and comply with this requirements. Major
improvements in FS have come mainly in VCOs and computer aided design tools of simulation
and design, and in a variety of digital techniques; speeds, densities, integration, and functionality.
Digital frequency technology attract high-level attention form system designers and industry leaders, for cellular and personal communications, radio and manufacturers, and government research and development (R & D) programs. The three FS technique do compete, but mainly complement one another, and most modern design utilizes combinations of technologies to achieve high performance, compactness, and economy.

Proliferation of digital and DSP discipline in the FS market has been taking place and more is expected. The designers must be familiar with analogue, digital, and radio frequency and microwave technologies to fully exploit their respective advantages.

Frequency synthesis is emerging as an exciting and dynamic field, with evolving challenges. New ground is broken often, and much attention is given to digital techniques in most technical meetings. FS is now an integral part of any complex system design since the enhanced capabilities of the FS offer much more than just the generation of sine waves, for modulation capabilities and complex waveforms can be designed as part of the FS.
CHAPTER THREE

PHASE LOCKED LOOP FREQUENCY SYNTHESIZER

3.1 **INTRODUCTION**

Chapter one dealt with the PLL frequency synthesizer in a summary form. This builds more on that, starting with phase locked loops and then delves into phase locked loop frequency synthesizers.

3.2. **PHASE LOCKED LOOP**

Phase locked loop is a circuit which synchronizes a signal from an oscillator to a reference signal usually from a crystal while the synchronized signal is from a local oscillator (usually a VCO). This is done through having a feedback kind of a circuit where by the two signals are fed into a phase comparator then a difference between the two signals made to drive the VCO. The diagram for this is as in figure 3.2a

![Figure 3.2a Basic phase locked loop](image)

Figure 3.2a Basic phase locked loop
The problem with the above phase locked loop is we have noise occurring in higher frequencies, so a low pass filter is put between the phase detector and the VCO. This results in a phase locked loop as in figure 3.2b.

![Figure 3.2b phase locked loop with loop filter](image)

Now the whole idea of a PLL FS is to generate frequencies which a crystal oscillator. But from the figures 3.2a and 3.2b, we have not achieved this, we have made possible for the VCO to lock into the crystal oscillator. Now if a way would be found to "fool" the VCO into generating a frequency or frequencies not generated by the quartz crystal, and then we have a FS.

A way was found to achieve that. Suppose the reference frequency in figure 3.2b is 20MHz after some time because of the feedback the VCO output frequency will also be 20MHz. Now if we divide the output frequency of the VCO before feeding to the PD by a number say 4, the phase detector will be comparing 20MHZ and 4MHZ. This will generate a large error that will force the VCO to raise its output frequency until the frequencies being compared are equal i.e. 20MHz. And for this to occur it has to generate 80MHz. This is a PLL frequency synthesizer.

### 3.3 Phase Locked Loop Applications

Due to low cost, low power consumption and stability in frequency generation PLL has became a common device in modern communication systems especially in the wireless arena. It finds application in the following places:

1) Frequency synthesizers
2) Modulation and Demodulation of FM signals

3) Modulation and Demodulation of AM signals.

4) FSK Decoders

5) Two-tone Decoders

6) Motor-speed Control

7) Frequency multiplications

3.4 **PHASE LOCKED LOOP FREQUENCY SYNTHESIZER.**

A phase locked loop frequency synthesizer is made up of the following basic components

1) Phase detector/comparator (PD/PC) and logic to charge converter

2) Loop filter (LF)

3) Voltage controlled oscillator (VCO)

4) Dividers

The type of a PLL FS depends on how division is implemented, resulting in two types of PLL FS: traditional also called integer- and fractional-N FSs. A basic PLL is shown in figure 2.3a
Figure 3.4 a simple PLL synthesizer

Phase detector compares the phase of the output signal, $\theta_o$, with the phase of the reference signal, $\theta_r$, and develops $\theta_e$. This $\theta_e$ turns on the logic to charge, (current) converter generates a current $ipd$ and starts to deliver a charge $qpd$ to the capacitor in the loop filter. At the end of a reference cycle, the charge delivered denoted as $qpd_{final}$, is designed to be linearly proportional to the phase error, $\theta_e$, as in figure 2.3b.

### 3.4.1 Phase Detector and Charge Pump

The purpose of the PD is to produce a signal that is proportional to the in phase between two signals. There are two main categories: analog and digital. The key characteristics are as follows:

1) Gain (sometimes associated with output or pump stage)

2) Linearity

3) Steering characteristics

There different types of PDs available but of special interest are:

1) Tristate phase/ frequency detector (PFD)
2) JK flip-flop based PD
3) EXOR gate based PD
4) Multiplier PD

For most PLL-based frequency synthesizers PFD is the preferred choice. For some other applications, other types of PD, such as EXOR and JK based PD can also be used.

**Phase Frequency Detector (PFD)**

Figure 3.4.1a is a PFD based on D flip-flop. PFD differs greatly from other type of phase detector. The major difference is the existence of the third state which leads to its major advantage over other types of PFDs. The PFD’s output signal depends not only on the phase error $\theta$, but also on the frequency error $\Delta f = F_{\text{ref}} - F_{\text{out}}$

$$\phi = F_{\text{ref}} \bar{\phi} F_{\text{out}}$$

Before locking is acquired AND resets when output goes to unused state. State and phase difference are defined as:

1) Up=0, Down=1, state = dpd = -1
2) Up=0, Down = 0, state = dpd = 0
3) Up=1, Down = 0, state = dpd = 1
4) Up = 1, Down = 1, state = dpd = inhibited
The actual state of the PFD is determined by the signals $V_r$ and $V_o$, figure 3.4.1b is a phase diagram of PFD.
Figure 3.4.1b a phase diagram of PFD.

**EXOR Phase Detector**

Figure 3.4.1c is an EXOR phase detector.
The EXOR differs from the PFD in that it has only two states (i.e. $d_{pd}$): -1 and 1 as opposed to three states of PFD.

**JK flip-flop detector**

Figure 3.4.1d is a JK flip-flop based PD
**Multiplier**

It is used exclusively in linear PLL, where sine wave is mostly used. Slow to acquire lock because of only comparing phases unlike PFD, which compares both the phase and the frequency. A Gilbert cell is an example of a multiplier.

**Charge pump**

A phase detector (both PFD and EXOR) needs logic to charge (current) converter or a charge pump. Figures of EXOR and PFD have charge pumps at the end.
3.4.2 Crystal Oscillator and Voltage Controlled Oscillator (VCO)

There are two places that the PLL loop contains an oscillator. The first oscillator is a crystal reference, which is a fixed, high quality source. The second is the voltage controlled oscillator (VCO), which translates a voltage to frequency.

Crystal oscillator can be Temperature Compensated Crystal Oscillator (TCXO), which has a temperature and compensation to correct the crystal frequency over temperature. The other is Oven Controlled Crystal Oscillator (OCXO).

Types of oscillator are summarized in the table 3.4.2 [9]

<table>
<thead>
<tr>
<th>Circuit type</th>
<th>Resonant circuit</th>
<th>Tuning range</th>
<th>Phase noise</th>
</tr>
</thead>
<tbody>
<tr>
<td>RC oscillator</td>
<td>Resistor and capacitor</td>
<td>Wide</td>
<td>Poor</td>
</tr>
<tr>
<td>Standard LC VCO</td>
<td>Inductors and capacitors</td>
<td>Wide</td>
<td>Fair</td>
</tr>
<tr>
<td>Strip line VCO</td>
<td>Microchip</td>
<td>Wide</td>
<td>Fair</td>
</tr>
<tr>
<td>SAW</td>
<td>SAW filter</td>
<td>Narrow</td>
<td>Excellent</td>
</tr>
<tr>
<td>VCXO</td>
<td>crystal</td>
<td>Very narrow</td>
<td>Best</td>
</tr>
<tr>
<td>CRO</td>
<td>Ceramic</td>
<td>Wide</td>
<td>Excellent</td>
</tr>
<tr>
<td>DRO</td>
<td>Dielectric</td>
<td>Wide</td>
<td>Excellent</td>
</tr>
<tr>
<td>YIG oscillator</td>
<td>YIG sphere</td>
<td>Very wide</td>
<td>Fair</td>
</tr>
<tr>
<td>Silicon VCO</td>
<td>Varies, but bond wires to implement L</td>
<td>Very wide</td>
<td>Fair</td>
</tr>
</tbody>
</table>

3.4.3 Loop filter.

Output of the phase detector consists of a number of terms; in locked state of PLL the first of these is a DC component and is roughly proportional to the phase error $\theta$; remaining terms are $\omega$ components having frequencies of $2 \times f_1, 4 \times f_1$. 
Because higher frequencies are unwanted they are filtered out by the loop filter. The loop filter passes low frequencies but suppresses higher, therefore it must be a low pass filter. In most PLL designs first order low pass filters are used. The loop filter determines a lot about the PLL performance:

1) Switching time

2) Loop bandwidth

3) Reference spurs

There are various types of loop filters:

a) **Passive lead-lag filter**

This filter has one pole and one zero, its transfer function is given by the equation

\[
F(s) = \frac{(1 + s \omega_2)}{1 + s(\omega_1 + \omega_2)}
\]

Where \( \omega_1 = R_1 C \) and \( \omega_2 = R_2 C \)

Zero has an important influence in damping factor \( \xi \) of the PLL system.

b) **Active lead-lag filter**

Its transfer function is similar to that of passive lead-lag filter but has an additional gain term \( K_a \) which can be chosen to be greater than 1. The transfer function is;

\[
F(s) = K_a \left[ \frac{(1 + s \omega_2)}{1 + s(\omega_1 + \omega_2)} \right]
\]

Where \( K_a = C_1 / C_2 \)

c) **Active PI filter**
It is a lead-lag filter whose transfer function is given by;

\[ F(s) = \frac{1 + s\omega}{s\omega} \]

Higher order loop filters can cause instability in the PLL therefore care should be taken when using them.

3.4.4 Dividers

Also called down-scalers and they come into play when the PLL is used as a frequency synthesizer. The down-scaler divides the output frequency created by the VCO by a factor N which is programmable in most cases. Down-scalers are usually built from a cascade of flip-flops (RS, JK or toggle).

A counter can scale down by an integer number only e.g. a factor of 10 or 12 not 10.5, nevertheless this is possible in fractional-N synthesizers and how this is done is discussed in fractional-N synthesizer section.

There are different categories of down-scalers, depending on the divider numbers possible. They include single modulus, dual modulus and quadruple modulus

3.5 KEY PARAMETERS OF A PLL SYNTHESIZER

PLL is governed by a set of key parameters that are

1) The lock range

This is the frequency range within which a PLL locks within one single beat note between reference and output frequency. Normally operating frequency range of a PLL is restricted to the lock range.

2) The pull-out range

This is the dynamic limit for stable operation of a PLL. If tracking is lost within this range, a PLL will normally lock again, but the process can be rather slow.
3) The pull-in range

This is the range within which a PLL will always be locked, but the process can be rather slow.

4) The hold range

This is the frequency range in which a PLL is able to maintain a lock. PLL locks out forever when the frequency of the input signal exceeds the hold range. This parameter is more of academic interest and most practitioners do not worry about its actual value. It's obtained by calculating that frequency where the phase error is at its maximum.

Additional parameters relate to time required for the PLL to get locked and are:

1) The lock time

This is the time the PLL needs to get locked when the acquisition process is a fast lock-in process

2) The pull-in time

The time the PLL needs to get locked when acquisition process is a slow pull-in process.
CHAPTER FOUR

INTEGER-N AND FRACTIONAL-N SYNTHESIZERS

4.1 INTRODUCTION

Integer-N is also called a traditional synthesizer, because it is based on the original FS which was to produce set frequency integer multiple of mostly a fixed reference. They are simple compared to fractional-N synthesizers and they are found in FM radio receiver, CB transceivers, TV receiver etc. Part one of this chapter deals with integer-N synthesizer showing its shortcomings and a way of going around them occasionally. Then part two delves into a more interesting and better performing fractional-N synthesizer

4.1.1 INTEGER-N SYNTHESIZER

Figure 4.1.1a is a basic integer-N synthesizer.

Figure 4.1.1a a simple integer-N synthesizer.

This frequency synthesizer is applied where there is a need for generating a great number of frequencies with narrow spacing of 50, 25, 10, 5, or even 1KHz. If the desired channel spacing is
10 KHz, a reference frequency of 10 KHz is normally chosen. But most oscillators are quartz crystal stabilized which are bulky in the kilohertz frequencies. It is therefore more convenient to generate reference frequencies in megahertz range like 5-10 MHz, and then scale it down to the desired reference frequency. This leads to having a divider before a PD giving us a PLL FS in figure 4.1.1b.

![ PLL Diagram ]

Figure 4.1.1b a simple Integer PLL with a reference divider.

When scaling factor of the reference frequency is denoted R and divider N, the VCO creates an output given by

\[ f_2 = \frac{Nf_1}{Rf_{osc}} = Nf_1 \]

One seeks to include as many functions as possible on the chip, and now it is to implement all digital functions on a chip such as oscillators, phase detectors, frequency dividers and so on. Because of its lower power consumption, high noise immunity, and large range of supply voltages, CMOS is the preferred technology today. The limited speed of CMOS precludes their application for directly generating frequencies in the range of 100MHz (that is in 2003). For generating higher frequencies, prescalers are used: this are built with other IC technology such as ECL, Schottky TLL, GaAs (gallium arsenide) or SiGe (silicon-germanium compound). Such
prescalers extend the range of frequencies into microwave frequency bands. This is shown in figure 4.1.1c. The output frequency of the synthesizer becomes

\[ F_{\text{out}} = NVf_1 \]

The scaling factor \( V \) is usually more than 1 in most circumstances, implying it is no longer possible to generate every desired integer multiple of the reference frequency \( f_1 \). If \( V = 10 \), only frequencies of \( 10f_1, 20f_1, 30f_1 \ldots \) can be generated. This can be circumvented by use of a so-called dual modulus prescaler, as shown in figure 4.1.1d. A dual-modulus prescaler is a counter whose division whose division ratio can be switched from one value to another by external control signal.

![Diagram](image)

Figure 4.1.1c an integer-N synthesizer with a prescaler

It can be shown that a dual-modulus prescaler makes it possible to generate a number of output frequencies that are spaced only by \( f_1 \) and not multiples of \( f_1 \).
The following conventions are used with respect to the dual prescaler PLL synthesizer

a) Both programmable \( \div N_1 \) and \( \div N_2 \) are down scalers

b) The output signal of both of these counters is high if the content of the corresponding counters has not yet reached to 0.

c) When the \( \div N_1 \) counter has counted to 0, its output goes low and immediately loads both counters to the preset values \( N_1 \) and \( N_2 \)

d) \( N_1 \) is always greater than or equal to \( N_2 \).

e) Due to AND gate, under flow below 0 is inhibited in the case of the \( \div N_2 \) counter. If has counted to 0, further counting pulses are inhibited.

The operation of the system becomes clear if we assume that the \( \div N_1 \) counter has just counted down to zero, and both counters are loaded with their present values \( N_1 \) and \( N_2 \) respectively. We have now the number of cycles the VCO must produce until the same logic state is reached again. This number is the overall scaling factor \( N_{\text{tot}} \) of the arrangement shown figure 4.4. As long as \( \div N_2 \) counter has not yet counted to 0, the prescaler is dividing by \( V+1 \). Consequently both \( \div N_1 \) and \( \div N_2 \) counters will step down by one count when the VCO has generated \( V+1 \) pulses. The
\( \div N_2 \) counter will therefore step down to zero when the VCO has generated \( N_2(V+1) \) pulses. At that moment the \( \div N_1 \) counter has stepped down by \( N_2 \) counts, i.e. its content is \( N_1-N_2 \). The scaling factor of the dual-modulus prescaler is now switched to the value \( V \). The VCO will have to generate additional \( (N_1-N_2)V \) pulses until the \( \div N_1 \) counter will step to 0. When the content of \( N_1 \) becomes 0, both \( N_1 \) and the \( N_2 \) counters are reloaded to their preset values, and the cycles is repeated.

How many pulses did the VCO produce in order to run through one full cycle?

\[
N_{\text{tot}} = N_2(V+1) + (N_1-N_2) \quad \text{factoring out yields}
\]

\[
N_{\text{tot}} = N_1V + N_2
\]

As stated, \( N_1 \) must always be greater than or equal to \( N_2 \). If not \( N_2 \) counter would be stepped down to 0 earlier than the \( \div N_2 \) counter and both counters would be reloaded to their preset values, the dual-modulus prescaler would never be switched from \( V+1 \) to \( V \), so the system would not work in the intended way.

If \( V=10 \), then using equation 4a

\[
N_{\text{tot}} = 10N_1+N_2
\]

In this expression, \( N_2 \) represents units and \( N_1 \) the tens of the overall division ratio \( N_{\text{tot}} \). Then \( N_2 \) must be in the range of 0 to 9, and \( N_1 \) can assume any greater value than or equal to 9; \( N_{\text{min}}=9 \).

The smallest division is therefore:

\[
N_{\text{tot}} = N_{1\text{min}}V = 90
\]

If we choose \( V=16 \), dual will divide by 16 or 17 then \( N_{1\text{min}} \) must be 15 or more.

Overall division ratio = 16(15) + 0 = 240

For \( V=100 \), the division ratio would be

\[
N_{\text{tot}} = 100N_1+N_2
\]

\( N_2 \) should be between 0 to 99 and \( N_{1\text{min}} \geq 99 \).
\[ N_{\text{tot(min)}} = 100(99) = 9900 \]

If reference frequency is 10 KHz lowest frequency to be synthesized is 99MHz.

There is another way, which extends the upper frequency range of the frequency synthesizer but still allows the synthesis of lower frequency; and is four-modulus prescaler which is a logical extension of the dual-modulus prescaler. It offers four different scaling factors, and two control signals are required to select one of the four available scaling factors. This is as in figure 4.1.1e

![Figure 4.1.1e four \( \bar{N} \) modulus integer-\( N \) synthesizer](image_url)

This extends the high end of frequency range while allowing lower frequencies than those obtainable from a synthesizer with dual-modulus prescaler. We have an internal circuitry to produce 100, 101 and 110, 111 controlled by two control signals, at first we have both low and
100 is in use, when the first control signal becomes high it changes to 101, it becomes 110 when another becomes high and for both high we have 111. We now have three programmable counters, in the system ÷N_1, ÷N_2 and ÷N_3.

\[ N_{tot} = 100N_1 + 10N_2 + N_3 \]

In this equation N_3 represents units, N_2 tens and N_1 hundreds of the division ratio. Here N_3 and N_2 must be in the range 0 to 9, and N_1 must be at least as large as both N_2 and N_3 for both reason explained in the previous example \(N_{1\text{min}} = 9\).

### 4.1.2 SHORTCOMINGS OF INTEGER-N SYNTHESIZER

Integer-N PLL synthesizer has the following short comings due to a necessary trade off in phase noise, frequency step-size or resolution, tuning speed or channel switching speed, spurs size and operating frequency range.

The first trade-off is between step size and tuning speed. Step size is tied to the reference frequency; however the loop bandwidth is limited by the reference frequency.

Another trade-off is between step size and phase noise. Phase noise is determined by the in-band phase noise floor and loop bandwidth. Phase noise can be lowered by narrowing loop bandwidth (i.e. at cost of a slower channel switching speed).

Lastly there is a trade-off between step-size and reference feedthrough spurs. Here the lower the reference the higher the reference feedthrough spurs if leakage is the dominating cause of spurs.

With all this trade-offs it is not a wonder that a fractional-N synthesizer came up.
4.2 FRACTIONAL-N SYNTHESIZER

4.2.1 INTRODUCTION

Fractional-N refers to a family of synthesizers that allow the minimum frequency step to be a fraction of the reference frequency. They are basically based on the integer-N structure with the presence of PD, LF, VCO, dividers etc. The difference is a fractional-N divider is not an integer as in the traditional PL case. We can have a divider number with a decimal part usually referred to as the fractional part. The benefits of these are enormous: fine frequency resolution, fast switching speed and a good noise performance [8].

In cases where high frequencies are to be generated in the gigahertz range and small channel spacing, traditional PLL performs poorly. This is due to the large divider number to be used and the small reference frequency, large divider N results in high phase noise and slow switching speed, while small reference results in a narrow loop bandwidth. To do away with these disadvantages we may use a multiloop PLL but this increases the complexity and the cost of the PLL.

The solution is to use a fractional-N PLL which power consumption is same as the traditional PLL but has the mentioned advantages. This section shows how the fractional part is achieved and how the benefits come about.

4.2.2 FRACTIONAL-N SYNTHESIZER TYPES.

The frequency synthesized by a fractional-N synthesizer can be a non-integer multiple of the reference frequency, as illustrated by the following equation:

$$ F_{\text{out}} = (N + k/M) \times F_{\text{ref}} $$

Where \( k \) and \( M \) are integers.

The variable \( M \) is a measure of the fractionality that a fractional-N synthesizer can provide. It is usually referred to as \( \text{fractional-modulus} \) or \( \text{fractional denominator} \). The integer \( k \) can assume any number between 0 and \( m \). The non-integer \( (N + k/M) \) is often written as \( N.F \), where the dot denotes decimal point and \( N \) and \( F \) represent the integer and fractional parts of the number respectively.
A number of methods have been proposed to implement fractional-N synthesizer among them the best known in the industry are these three:

a) Fractional divider-based

b) Current injection-based

c) Delta-sigma (ΔΣ) modulator-based

The first method is based on the traditional PLL while the other two on the concept of division ratio averaging.

Fractional Divider-Based Fractional-N

This method evolved from the basic principles of integer-N synthesis. The only difference is that the frequency divider is replaced with a fractional divider. A fractional frequency divider is no longer a simple digital counter. The period of the divider is given by the following equation:

\[ T_{do} = (N + 0.F) \times T_{vco} \]

where 0.f = a fractional number

\[ T_{vco} \] = the period of the VCO

Once N and 0.F are set, the period of a fractional divider output is ideally not time varying. In other words, a rising age occurs at the output each N and 0.F cycle. As with integer-N synthesizer, \( T_{do} \) is forced to follow the reference period. Therefore,

\[ T_{ref} = (N + 0.F) \times T_{vco} \] or

\[ f_{vco} = (N + 0.F) \times f_{ref} \]

\( T_{ref} \) is the reference period and \( f_{ref} \) the reference frequency.

If a fractional divider block is dual, it will have: a dual modulator divider (DMD), a delay locked loop (DLL), a multiplexer (MUX), and a digital phase accumulator (DPA), though a fractional divider does not have to be based on DLL.
The DLL consists of a set of cascaded delay elements; a phase detector; a charge pump; and a D-flip flop. The negative nature of the feedback in the DLL ensures that the total delay through the delay line is one VCO cycle. Since the delay elements are, ideally, identical, a VCO period is broken up into $N_d$ equal packets of phase, where $N_d$ is the number of delay elements in the delay line.

A simple DPA is made up of an adder and a register. The register is clocked by a reference clock. The input to the DPA is an $m$-bit word. The contents of the register are used to control the MUX. On every reference rising edge, the contents are incremented by the value of the input, $x$, which is represented by an $m$ bits word. The output of the DPA (i.e. the carry-out of the adder) is one-bit quantization of the input.

The number of bits in the accumulator ($m$) is related to the number of discrete packets of phase by the following equation:

$$N_d = 2^m$$

The output of the DPA controls the DMD. When carry-out is high, the DMD divides by $N+1$ and when low $N$. In the following example, the fractional division ratio, $N + 0.F$, for a DPA input $x$, is equal to $N + x/2^m$.

Suppose the DPA has three bits and, therefore, the delay line has eight elements. Each phase packet corresponds to 1/8 of the VCO cycle. Also assume that the input is equal to 2, which corresponds to a 0.F of 2/8. When no carry-out occurs, the DMD divides by $N$. Its output however is not immediately presented to the PFD of the PLL. Rather, it is delayed by a number of phase packets controlled or selected by the MUX content of the DPA, which is incremented by 2 every reference cycle. This means that the output is phase-shifted by a progressively increasing number of phase packets (i.e., 0, 2, 4, 6, 8) each reference cycle. As a result, the period of the DMD output is increased by 2/8 of the VCO cycle. Therefore, the effective division ratio becomes $N + 0.25$, which is what it should be.

When the DPA content reaches 8, the content of the DPA is reset, and the output of the DMD is not delayed by the delay line. However, this coincides with a carry-out, which forces the DMD
to divide by N+1. This is equivalent to the DMD dividing by N and its output being delayed by 8 phase packets (i.e. one VCO cycle).

The design of the fractional divider dictates the fractional modulus or denominator to be \( N_d \), the number of delay elements. Because all the elements in the delay line operate at the VCO speed, the added power consumption can be significant, especially when the VCO frequency and or fractionality is high. Another drawback of this method is that the edges of the fractional divider output may be noisy as a result of jitter on the outputs of the delay elements. This jitter is because of the mismatch and phase error from the phase error-correcting action of the DLL. The edge contamination may result in a significant increase in the phase detector noise floor. [8]

**Averaging Fractional-N**

Another way to achieve fractional-N synthesis is by division ratio averaging. The idea is use of an integer divider as opposed to the fractional divider, but the division ratio dynamically switched between two or more values. The result is division by a non-integer number. The number determined by the values which the division ratio is changed and probability of each ratio used.

For example, if the divider divides by 50 half of the time and 51 the other half, the average division ratio is 50.5. In general, the non-integer division ratio is given by the equation:

\[
N.F = N_1P_1 + N_2P_2 + \ldots + N_jP_j
\]

Where \( N.F \) is the average division ratio, and \( N_j \) and \( P_j \) are the integer division ratio and the probabilities associated with them, respectively. Again, since the average divider output frequency is equal to \( f_{\text{ref}} \), the following relationship exists when the loop is in lock:

\[
f_{\text{vco}} = N.f \times f_{\text{ref}}
\]

One way to switch the division ratio dynamically is through the use of a simple modulus controller. A modulus controller can be a simple DPA. The output of the DPA is used to control the division ratio of a DMD. The divider divides by \( N + 1 \) when there is a carry-out and by \( N \), otherwise. In this case, the fractional part of the average division ratio is equal to the input of the DPA.
DPA like any ΔΣ modulator exhibits quantization error. Quantization error and how it results in quantization phase error and how to do with them are discussed in the following section

**Quantization phase error**

The discussion that follows is based on a DPA-controlled, dual-modulus divider

Quantization error in the DPA output exists because the output is an approximation of the input and is never equal to the desired value. This is simply because the output is either 0 or 1, while the input is between 0 and 1 (excluding 0 and 1).

Accordingly, the instantaneous division ratio of the DMD is never equal to the average ratio by which the imaginary divider divides. This, in turn gives rise to a phase difference between the actual instantaneous DMD output and the imaginary divider output. If the latter is viewed as a reference, then the former exhibits phase error with respect to the latter. This phase error is defined as quantization phase error or quantization phase noise. It is referred to us quantization noise for simplicity.

Two points are worth mentioning. First, the imaginary divider output is phase-locked to the reference. Phase difference between the two signals is such that there is no error correction signal at the output of the charge pump. Secondly, the waveform of the actual DMD output can be viewed as the imaginary divider output with its phase being modified by the quantization phase error.

Quantization errors cause phase errors at PFD. The waveform of the phase error sensed by the PFD is the quantization phase error with, possibly some DC offset. The phase error at the PFD gives rise to a current signal at the charge pump output. The current waveform is scaled version of the phase error with the scaling factor being the charge pump gain. The current is converted to a voltage signal by the loop filter that modulates the instantaneous frequency of the VCO.

If the quantization phase error waveform exhibits some periodicity and if the magnitude of the waveform is large, it shows up in the voltage signal as well. As a consequence the VCO is modulated periodically, which causes spurs in the VCO spectrum at the offset frequency corresponding to the periodicity of the current waveform and its harmonics.
**Current Injection-Based Fractional Compensation**

Fractional spurs mentioned averaging fractional-N synthesis is a serious problem. The magnitude of the periodic waveform of the resulting quantization phase error waveform is large compared to random jitters on the DMD output or the reference fractional spurs typically only 20 to 30 dB below the carrier. As a consequence various methods of suppressing these errors have been devised.

It has been noticed that the quantization error change with well defined amount every reference cycle. If a current pulse train can be injected with the same width but opposite sign to the integrating capacitor in the loop filter, the quantization phase error can be ideally cancelled. This is called current injection-based fractional compensation from which we get the name of another type of fractional-N synthesizer.

But usually, mismatches exist between the amplitude and the width of the compensation current and those of the charge pump, and therefore, cancellation of quantization phase error is imperfect. Therefore fractional spurs can still be quite large. [8]

**Delta Sigma Fractional-N PLL**

Though it is theoretically possible to use analogue compensation schemes to completely eliminate the fractional spurs without any ill effect, many issues arise in practical applications. Schemes for current injection tend to be difficult to optimize to account for variations in process, temperature and voltages, while schemes involving time delay tend to add phase noise. Also as the fractional modulus gets large, the analogue compensation architectures become more complex. Another way of reducing the fractional spurs was devised; delta sigma modulation.

Delta sigma PLL have no analogue compensation and reduce fractional spurs using digital techniques. This is done through alternating the N counter between more than two values. The impact on the frequency spectrum is it pushes the fractional spurs to higher frequencies that can be filtered by the loop filter. For example consider a PLL with N value 100.25 and compensation clock of 1 MHz a traditional fractional PLL will achieve this by alternating between 100 and 101. Second order delta sigma modulator would achieve this by alternating the N counter between the values 98, 99, 100 and 101. Table 4.2.2 shows the order and inputs required.
Table 4.2.2 delta sigma order and the inputs

<table>
<thead>
<tr>
<th>Delta sigma order</th>
<th>Delta sigma inputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1&lt;sup&gt;st&lt;/sup&gt;</td>
<td>0, 1</td>
</tr>
<tr>
<td>2&lt;sup&gt;nd&lt;/sup&gt;</td>
<td>-2, -1, 0, 1</td>
</tr>
<tr>
<td>3&lt;sup&gt;rd&lt;/sup&gt;</td>
<td>-4, -3, -2, -1, 0, 1, 2, 3</td>
</tr>
<tr>
<td>k&lt;sup&gt;th&lt;/sup&gt;</td>
<td>-2&lt;sup&gt;k&lt;/sup&gt;, ê ê ê ê , 2&lt;sup&gt;k&lt;/sup&gt;-1</td>
</tr>
</tbody>
</table>

The sequence generated by the modulator depends on the modulator order.

Spurs can also reduced in this type of PLL by dithering i.e. having randomness in the sequence generated. [9]

4.2.3 ADVANTAGES OF FRACTIONAL-N PLL

In terms of performance parameters, the important parameters of a synthesizer include phase noise, frequency step or frequency resolution, tuning speed or channel switching speed, spur size, and operating frequency range.

Other important parameters include current consumption, power supply voltage, cost, package size, pin count, etc.

Fractional-N PLL strengths are integer-Ns weaknesses. While the reference frequency for the traditional PLL can not be more than the channel spacing in order to synthesize all the required frequencies, this is not so with fractional-N PLL. This is due to the ability of the divider of fractional-N PLLs to have a decimal part enabling it when multiplied with higher frequencies to still give the desired frequency. The advantage of a high reference frequency is reduction of the
phase noise and a higher tuning speed. This is because phase noise in a PLL is given by the formula;

$$20\log N \text{ where N is the divider value}$$

N in fractional-N synthesizers reduces several times due to a high frequency used resulting in a lower phase noise.

The tuning speed depends on the loop filter design for the bandwidth allowed. A narrow bandwidth results in slow tuning speed, while broader allows for a faster tuning speed. The bandwidth depends on the reference frequency, high reference frequency allows for a broad bandwidth.

Fractional-N PLL also makes it possible to have high frequency resolution, compared to integer-N PLL. Integer-N PLL’s frequency of resolution is same as its channel spacing while fractional-N PLL’s is given by the formula;

$$\Delta f = \left(\frac{1}{M}\right) \times f_{\text{ref}} \quad [10]$$

Where $\Delta f$ is frequency resolution, M is the fractional-modulus and $f_{\text{ref}}$ is the reference frequency.

### 4.2.4 SUMMARY

Fractional-N synthesizers allow the frequency step size to be a fraction of the reference frequency. This makes it possible to achieve low phase noise, small step size, fast tuning and minimal reference feedthrough spurs at the same time.

A number of ways are there to achieve fractional-N synthesis and so far $\Delta \Sigma$ modulator-based PLL has emerged as the most successful technique to achieve all performance requirements at once.

Fractional spurs can still be a problem in $\Delta \Sigma$ PLL but normally this problem persists only over very narrow and predictable bands.
CHAPTER FIVE

DESIGN AND ANALYSIS

5.1 DESIGN PROBLEM

Designing a phase-locked loop based frequency synthesizer for use in the 12GHz range, with channel spacing of 36 MHz

5.2 DESIGN VALUES

A reference frequency of 360MHz was chosen because the phase noise at this frequency is less if I were to use the channel spacing as a reference frequency.

Because of the advantages of a fractional-N synthesizer, its design was chosen.

The divider numbers were calculated using the formula

\[ N.F = \frac{F_{\text{out}}}{F_{\text{ref}}} \]

Where N.F is the divider number, integer (N) and fraction (F)

\[ F_{\text{out}} \] is the VCO output frequency and \[ F_{\text{ref}} \] is reference frequency

The reference frequency is fixed and we know the frequency to be synthesized is already known. Calculating divider values the following design Values in table 5.2 were found.
Table 5.2 divider design values.

<table>
<thead>
<tr>
<th>Channel</th>
<th>Integer part (N)</th>
<th>Fractional part (F)</th>
<th>Synthesized frequency (GHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>33</td>
<td>333</td>
<td>12.000</td>
</tr>
<tr>
<td>2</td>
<td>33</td>
<td>433</td>
<td>12.036</td>
</tr>
<tr>
<td>3</td>
<td>33</td>
<td>533</td>
<td>12.072</td>
</tr>
<tr>
<td>4</td>
<td>33</td>
<td>633</td>
<td>12.108</td>
</tr>
<tr>
<td>5</td>
<td>33</td>
<td>733</td>
<td>12.144</td>
</tr>
<tr>
<td>6</td>
<td>33</td>
<td>833</td>
<td>12.180</td>
</tr>
<tr>
<td>7</td>
<td>33</td>
<td>933</td>
<td>12.216</td>
</tr>
<tr>
<td>8</td>
<td>34</td>
<td>033</td>
<td>12.252</td>
</tr>
<tr>
<td>9</td>
<td>34</td>
<td>133</td>
<td>12.288</td>
</tr>
<tr>
<td>10</td>
<td>34</td>
<td>233</td>
<td>12.324</td>
</tr>
<tr>
<td>11</td>
<td>34</td>
<td>333</td>
<td>12.360</td>
</tr>
<tr>
<td>12</td>
<td>34</td>
<td>433</td>
<td>12.396</td>
</tr>
<tr>
<td>13</td>
<td>34</td>
<td>533</td>
<td>12.432</td>
</tr>
<tr>
<td>14</td>
<td>34</td>
<td>633</td>
<td>12.468</td>
</tr>
<tr>
<td>15</td>
<td>34</td>
<td>733</td>
<td>12.504</td>
</tr>
<tr>
<td>16</td>
<td>34</td>
<td>833</td>
<td>12.540</td>
</tr>
<tr>
<td>17</td>
<td>34</td>
<td>933</td>
<td>12.576</td>
</tr>
<tr>
<td>18</td>
<td>35</td>
<td>033</td>
<td>12.612</td>
</tr>
<tr>
<td>19</td>
<td>35</td>
<td>133</td>
<td>12.648</td>
</tr>
<tr>
<td>20</td>
<td>35</td>
<td>233</td>
<td>12.684</td>
</tr>
<tr>
<td>21</td>
<td>35</td>
<td>333</td>
<td>12.720</td>
</tr>
<tr>
<td>22</td>
<td>35</td>
<td>433</td>
<td>12.756</td>
</tr>
<tr>
<td>23</td>
<td>35</td>
<td>533</td>
<td>12.792</td>
</tr>
</tbody>
</table>
5.3 DESIGN CIRCUIT

![Fractional-N PLL circuit diagram](image)

Figure 5.3 Fractional-N PLL circuit
**Pulse generator**

The Pulse Generator block generates the reference signal. The block produces a periodic pulse train. The variable $F_{REF}$, initially set 360 MHz, denotes the frequency of the pulse train.

The period of the pulse train is $1 / F_{REF}$.

Thus the pulse generator block generates the frequency which is a periodic pulse train.

**Gains**

A Gain block multiplies the output signal from the Analog Filter Design block by a constant to produce the control signal. The Gain parameter for this block is set to $(synFr*[(synN+synM) - synFq]) * 2 / synSen$.

For the default values of the variables, the gain is equal to 7/2. Thus, in the steady state of the model, the output of the Gain block is approximately constant, with a value of 7/4.

**Voltage-Controlled Oscillator**

The Continuous-Time VCO block generates the synthesized signal (along with the Convert to Square Wave subsystem) and adjusts the frequency of the synthesized signal according to the Voltage-Controlled Oscillator input signal.

When the control signal is close to its steady-state value of 7/4, the Continuous-Time VCO block generates a signal whose frequency is close to $synFr*(synN + synM)$. If the output frequency
drops, the control signal rises, boosting the frequency of the output signal. If the output frequency rises, the control signal falls, lowering the output frequency.

The Quiescent frequency parameter is just the oscillation frequency, synFq. The difference between the block's output signal frequency and the quiescent frequency is proportional to the input signal, interpreted as voltage. The quiescent frequency is set to the variable synFq, assigned a value of 12GHz.

The Input sensitivity parameter scales the input voltage, and thus controls the shift from the quiescent frequency. The units of the parameter are hertz per volt. The input sensitivity is set to the variable synSen, which is initially assigned a value of 36 MHz/V.

Changing the values of synFq and synSen will not affect the steady-state frequency of the synthesized signal, because the corresponding change to the gain value exactly compensates for the change.

**Accumulator**

The Accumulator repeatedly adds the constant synM to a cumulative sum. While the sum is less than 1, the output labeled Carry is 0. At a time step when the sum becomes greater than or equal to 1, the carry output is 1 and the cumulative sum is reset to its fractional part. The fraction of the time when the carry output is 1 is equal to synM, while the fraction of the time when it is 0 is equal to 1-synM*.

**Divide Frequency**

The Divide Frequency divides the frequency of the synthesized signal by synN when the output of the Accumulator subsystem is 0, and divides it by synN+1 when the output is 1. As a result, the average amount that frequency is divided by is

\[(1-synM)*synN + synM*(synN+1) = synN + synM\]
The line leading out of the Divide Frequency is labelled Divided synthesized.

**Phase Detector.**

The Logical Operator block acts as a phase detector. It applies the EXOR operation to the frequencies of the reference signal and the frequency of the output from the Divide Frequency subsystem.

**5.4 RESULTS AND DISPLAYS**

When the circuit was simulated, scope window displayed the following parameter as in figure 5.4a

![Figure 5.4a scope of the circuit during simulation.](image-url)
The scope at VCO output at divider values N=33 and M=.333 is shown in the figure 5.4b

Figure 5.4b VCO output at divider values N=33 and M=.333
The scope at VCO output at divider values $N=35$ and $M=0.233$.

Figure 5.4C VCO output at divider values $N=35$ and $M=0.233$.
Figure 5.4e VCO output at divider values N=36 and M
CHAPTER SIX

CONCLUSION AND RECOMMENDATIONS

A study of frequency synthesis was done with emphasis on phase locked loop based frequency synthesizers. Then a fractional-N frequency synthesizer was designed in the frequency range of 12 GHz and 36 MHz channel spacing. The designed circuit was simulated using MATLAB.

Recommendations

1) A further study in fractional spurs to get a more effective way of minimizing them

2) A study of SiGe synthesizers

3) Implementation of the design using SiGe BiCMOS technology
REFERENCES


