A Digital Frequency Synthesizer Using Phase Locked Loop Technique

A Thesis

Presented in Partial Fulfillment of the Requirements for

the Degree Master of Science in the

Graduate School of The Ohio State University

By

Gursharan Reehal, BSEE

* * * * *

The Ohio State University

1998

Master's Examination Committee:

Steve Bibyk, Adviser

Mohammed Ismail

Approved by

Adviser

Department of Electrical Engineering

ABSTRACT

Phase Locked Loops are used in almost every communication system. Some of its uses include recovering clock from digital data signals, performing frequency, phase modulation and demodulation, recovering the carrier from satellite transmission signals and as a frequency synthesizer. A frequency synthesizer is a circuit design that generate a new frequency from a single stable reference frequency. Mostly a crystal oscillator is used for the reference frequency. Most of the frequency synthesizer employ a Phase Locked Loops circuit, as this technique offer many advantages such as minimum complex architecture, low power consumption and a maximum use of Large Scale Integration technology. There are many designs in communication that require frequency synthesizer to generate a range of frequencies; such as cordless telephones, mobile radios and other wireless products. The accuracy of the required frequencies is very important in these designs as the performance is based on this parameter. One approach to this necessity could be to use crystal oscillators. It is not only impractical, but is impossible to use an array of crystal oscillators for multiple frequencies. Therefore some other techniques must be used to circumvent the problem. The main benefit of using Phase Locked Loop technique in frequency synthesizer is that it can generate frequencies comparable to the accuracy of a crystal oscillator and offer other advantages mentioned previously. For this reason most of the communication design make use of a PLL frequency synthesizer. Considering the scope of this single circuit,

this Thesis is devoted to the research of a digital PLL frequency synthesizer. Phase locked loop is an excellent research topic as it covers many disciplines of electrical engineering such as Communication Theory, Control Theory, Signal Analysis, Noise Characterization, Design with transistors and op-Amps, Digital Circuit design and non-linear circuit analysis.

A Digital Frequency Synthesizer Using Phase Locked Loop Technique

By

Gursharan Reehal, M.S.

The Ohio State University, 1998

Steve Bibyk, Adviser

Phase Locked Loops are used in almost every communication system. Some of its uses include recovering clock from digital data signals, performing frequency, phase modulation and demodulation, recovering the carrier from satellite transmission signals and as a frequency synthesizer. A frequency synthesizer is a circuit design that generate a new frequency from a single stable reference frequency. Mostly a crystal oscillator is used for the reference frequency. Most of the frequency synthesizer employ a Phase Locked Loops circuit, as this technique offer many advantages such as minimum complex architecture, low power consumption and a maximum use of Large Scale Integration technology. There are many designs in communication that require frequency synthesizer to generate a range of frequencies; such as cordless telephones, mobile radios and other wireless products. The accuracy of the required frequencies is very important in these designs as the performance is based on this parameter. One

approach to this necessity could be to use crystal oscillators. It is not only impractical, but is impossible to use an array of crystal oscillators for multiple frequencies. Therefore some other techniques must be used to circumvent the problem. The main benefit of using Phase Locked Loop technique in frequency synthesizer is that it can generate frequencies comparable to the accuracy of a crystal oscillator and offer other advantages mentioned previously. For this reason most of the communication design make use of a PLL frequency synthesizer. Considering the scope of this single circuit, this Thesis is devoted to the research of a digital PLL frequency synthesizer. Phase locked loop is an excellent research topic as it covers many disciplines of electrical engineering such as Communication Theory, Control Theory, Signal Analysis, Noise Characterization, Design with transistors and op-Amps, Digital Circuit design and non-linear circuit analysis.

This work is dedicated to my Dear Mom and Dear Brother

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Steven Bibyk, for his assistance and guidance in not only the completion of this work and the determination of a topic for my research, but also for allowing me the freedom to work independently. In addition I would like to thank Dr. Mohammed Ismail for his time in serving on my committee.

VITA

February 02, 1972	Born - New Delhi, INDIA
1996	B.S. Electrical Engineering
1998	M.S. Electrical Engineering
Autumn 1996- Summer 1997	Graduate Adminstrate Associate, Ohio State University.
1997-present	Graduate Research Associate, Ohio State University.

FIELDS OF STUDY

Major Field: Electrical Engineering

TABLE OF CONTENTS

Page

LIST OF TABLES

Table **Page**

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Phase Locked Loops (PLL) are a new class of circuit, used primarily in communication applications. It is suitable for a wide variety of applications, such as AM radio receivers, frequency demodulators, multipliers, dividers, and as frequency synthesizers. The phase locked loops was first described in early 1930s, where its application was in the synchronization of the horizontal and vertical scans of television. Later on with the development of integrated circuits, it found uses in many other applications. The first PLL ICs came in existence around 1965, and was built using purely analog devices. Recent advances in integrated circuit design techniques have led to an increased use of the PLL as it has become more economical and reliable. Now a whole PLL circuit can be integrated as a part of a larger circuit on a single chip. This chapter gives a brief introduction to the basics of Phase Locked loops.

1.1 Phase Locked Loop Fundamentals

A Phase Locked Loop or a PLL is a feedback control circuit. As the name suggests, the phase locked loop operates by trying to lock to the phase of a very accurate input signal through the use of its negative feedback path. A basic form of a PLL consists of three fundamental functional blocks namely

- A Phase Detector (PD)
- A Loop Filter (LF)
- A voltage controlled oscillator (VCO)

with the circuit configuration shown in Figure 1.1

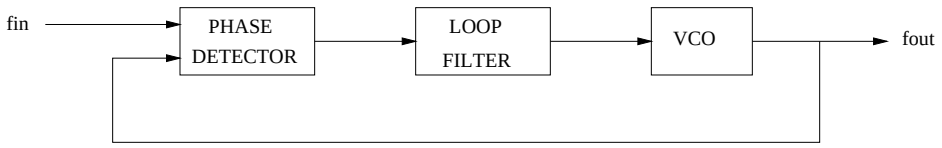

Figure 1.1: A basic Phase Locked Loop

The phase detector compares the phase of the output signal to the phase of the reference signal. If there is a phase difference between the two signals, it generates an output voltage, which is proportional to the phase error of the two signals. This output voltage passes through the loop filter and then as an input to the voltage controlled oscillator (VCO) controls the output frequency. Due to this self correcting technique, the output signal will be in phase with the reference signal. When both signals are synchronized the PLL is said to be in lock condition. The phase error between the two signals is zero or almost zero at this.

As long as the initial difference between the input signal and the VCO is not too big, the PLL eventually locks onto the input signal. This period of frequency acquisition, is reffered as pull-in time, this can be very long or very short, depending on the bandwidth of the PLL. The bandwidth of a PLL depends on the characteristics of the phase detector (PD), voltage controlled oscillator and on the loop filter. Before

going to look at overall loop operation, let us discuss the three main functional blocks in some more detail.

1.2 Phase Detector Overview

The role of a Phase Detector/comparator in a phase-locked loop circuit is to provide an error signal which is some function of the phase error between the input signal and the VCO output signal. Let θ_d represents the phase difference between the input phase and the VCO phase. In response to this phase difference the PD produces a proportional voltage v_d. The relation between voltage v_d, and the phase difference θ_d is shown in figure 1.2. The curve is linear and periodic, it repeats every 2π radians. This periodicity is necessary as a phase of zero is indistinguishable from a phase of 2π.

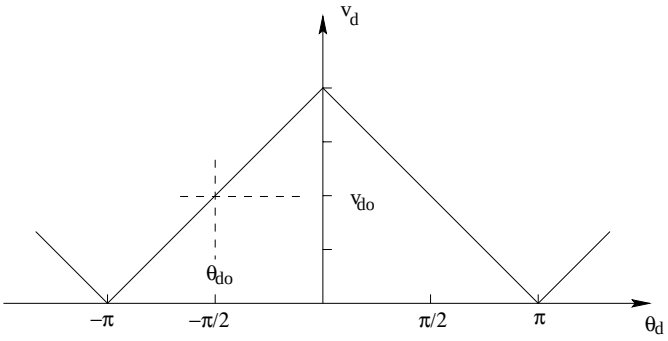

Figure 1.2: Phase Detector characteristics

In this general model of a phase detector,if no input is applied to PD, (or in other words when phase difference is zero between the two inputs of a PD) it generates a free running voltage v_{do}, which is shown as 4 volts in figure 1.2. Corresponding to

this v_{do}, there is a phase θ_{do} associated with it, which is $\pi/2$ as shown in the Figure 1.2.. The common approach is that a phase difference of zero should correspond to the free running voltage v_{do} of the PD. Thus, considering this approach the phase error can be defined as

$$\theta_e = \theta_d - \theta_{do} \tag{1.1}$$

and the shifted chractersic of the phase detector is shown in Figure 1.3

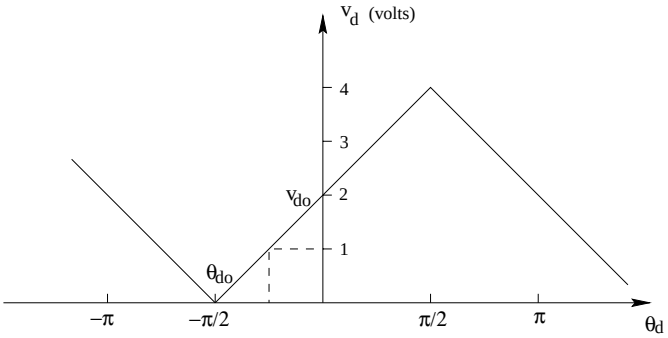

Figure 1.3: Phase Detector's shifted characteristics

The characteristic of PD is linear between $-\pi/2$ and $\pi/2$. The slope of the curve is constant and is equal to

$$K_d = \frac{dv_d}{d\theta_e} \tag{1.2}$$

So for the above case $K_d = 4v/\pi(radian) = 2.54v/rad$. The general model of a PD, thus can be represented with the following equation

$$v_d = K_d\theta_e + V_{do} \tag{1.3}$$

Based on this equation, the signal flow diagram of the PD is shown in the Figure 1.4 There are many ways to implement a Phase Detector circuit, but the most common

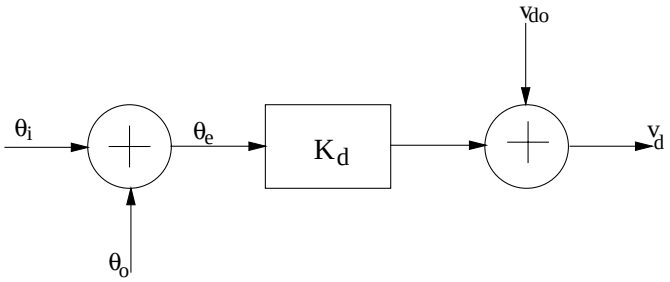

Figure 1.4: Signal Flow Model of Phase Detector

approach is the multiplying phase detectors. For multiplying phase detectors, the actual phase detector characteristics will depend on the waveforms of the input signals. For example two sinusoidal signals result in a sinusoidal PD chracterstics while two square wave signals generate a triangular charactersitcs. The three most important multiplying digital phase detectors are the following.

- The EXOR gate
- JK flip flop
- Phase-Frequency detector(PFD)

The underlying principle behind the operation of each of the phase detectors mentioned is the multiplication of the VCO signal with the input signal, which outputs a dc error signal that is a function of the phase error.

The other most commonly used technique to implement a phase detector is the sequential phase detectors. These types of PDs are constructed using digital circuit components, such as flip flops, latches and inverters. The only limitation for these types of phase detectors is the switching speed of the digital logic circuitry they employ. Therefore these types of PD are limited to lower rate frequency applications.

There are many other types of PD that are in use, but the ones mentioned above are the most commonly used for most of the applications.

1.3 VCO Overview

A VCO is a voltage controlled oscillator, whose output frequency ω_o is linearly proportional to the control voltage v_c generated by the Phase detector. This linear relation between the control voltage and the output frequency simplifies the PLL design. A typical characteristic of a voltage-controlled oscillator is shown in figure 1.5

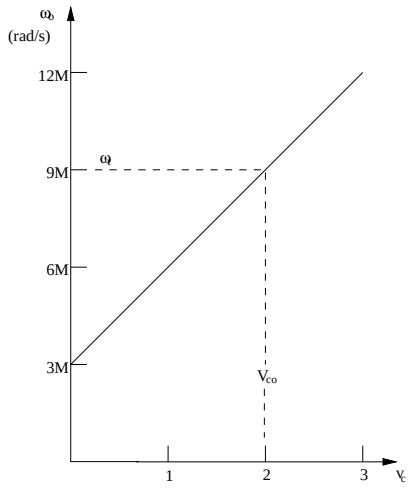

Figure 1.5: VCO characterstics

The slope of the curve is constant. As the v_c varies from 0 to 2 volts, the output frequency of the VCO varies from 3 Mrad/s to 12 Mrad/s. Outside this range the curve may not be linear and the VCO performance degrades or become non-linear. Depending on the specific requirements of a circuit, the range can be selected such

that the circuit always remain in its linear range, so the non-linear range is not an issue here. When the PLL is in the lock condition, the output frequency $\omega_o = \omega_i$. For an example suppose the output frequency of the VCO (ω_i) is 6 Mrad/s, from Figure 1.5, this frequency requires that the control voltage v_c should be 1 Volts. Which means $v_d = 1$ volts. A $v_d = 1$ requires a phase error of $\theta_e = -0.79$ rad. This average value of the phase error is called the static phase error.The basic approach is that the static phase error should remain near zero and must not increase beyond the PD linear range of$\pm\pi/2$ radians. Based on these constraints, the general srategidy is that v_c should correspond to $\Delta\omega_o$, the difference between ω_o and ω_i. This results in a shifted chracterstic of the VCO as shown in Figure 1.6

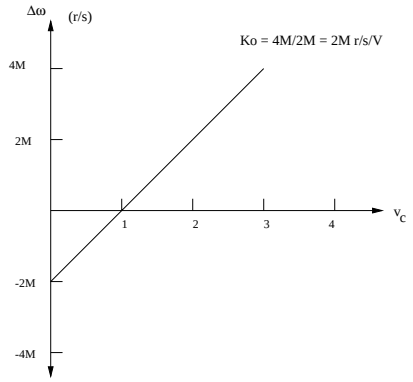

Figure 1.6: VCO's shifted characteristics

The plot is $\Delta\omega_o$ vs v_c. So $\Delta\omega_o = 0$ corresponds to $v_c = V_{co}$. The slope of this curve is the VCO gain K_o and is given by

$$K_o = \frac{d\Delta\omega_o}{dv_c} \tag{1.4}$$

The general model of the VCO is thus given as

$$\Delta\omega = K_o(v_c - V_{co}) \tag{1.5}$$

and the signal flow diagram is shown in Figure 1.7

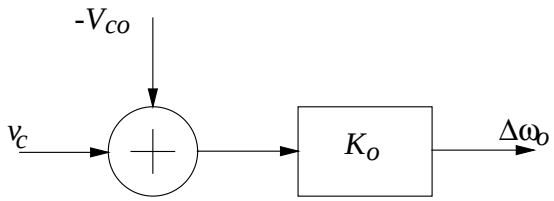

Figure 1.7: Signal Flow Model of VCO

where V_{co} is the control voltage, when PLL is in lock.

1.4 Loop Filter

The filtering operation of the error voltage (coming out from the Phase Detector) is performed by the loop filter. The output of PD consists of a dc component superimposed with an ac component. The ac part is undesired as an input to the VCO, hence a low pass filter is used to filter out the ac component. Loop filter is one of the most important functional block in determining the performance of the loop. A loop filter introduces poles to the PLL transfer function, which in turn is a parameter in determining the bandwidth of the PLL. Since higher order loop filters offer better noise cancelation, a loop filter of order 2 or more are used in most of the critical application PLL circuits.

1.5 PLL Bandwidth and Overall Loop Operation

The bandwidth of a PLL, which determines that how fast a PLL will be in following the input phase, or for how long it will remain in the lock condition, depends on the characteristics of the Phase detector (PD), the voltage controlled oscillator (VCO) and on the Loop filter. Since the bandwidth is associated with the ac model of a PLL, we can form an ac model by eliminating the dc parameters. The model is shown in Figure 1.8

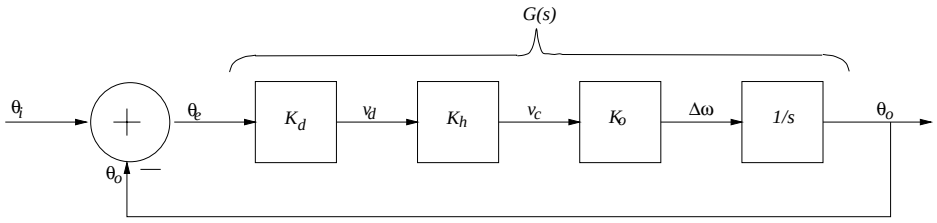

Figure 1.8: Linear Model of a PLL

The integration can be replaced with $1/s$, using Laplace transform, where s represents complex frequency. The closed loop transfer function may be found by applying Mason's rule, which for a single loop control system, such as this one reduces to the following simple formula.

$$Closed\ Loop\ transfer\ function = \frac{path\ gain\ from\ input\ to\ output}{1 - loop\ gain}$$

Applying this to signal flow model of PLL, we get

$$\frac{\theta_{r(s)}}{\theta_{o(s)}} = \frac{G(s)}{1 + G(s)} = \frac{G(j\omega)}{1 + G(j\omega)} \tag{1.6}$$

Where

$$G(s) = \frac{K_d K_h K_o}{s} \tag{1.7}$$

So the forward gain of the pll by is given by

$$K = K_d K_h K_o \tag{1.8}$$

Therefore the bandwidth ω_{3dB} occurs when $|G(j\omega)| = 1$ From the relation, this occurs when $1 = K/\omega_{3dB}$, or in other words when

$$\omega_{3dB} = K = K_d K_h K_o \tag{1.9}$$

The bandwidth of the PLL is thus determined by

- the gain K_d of the PFD
- the high frequency gain K_h of the loop filter
- and the gain K_o of the VCO

The designs of VCO and PD are usually less flexible, the design of the loop filter is the principle tool in selecting the bandwidth of the PLL [8]. The selection of loop bandwidth forces trade–offs in the frequency acquisition speed. Since PLL pull in speed is a function of the loop bandwidth, the simplest method of improving the frequency acquisition chracterstic is to widen the bandwidth of the loop filter. The wider the loop bandwidth, the faster the frequency acquisition will occur. However, the wider bandwidth degrades the tracking abilities of the PLL and increases the timing to obtain the desired output frequency, in the design of a frequency synthesizer.

CHAPTER 2

FREQUENCY SYNTHESIZER

One of the most common use of a PLL is in Frequency synthesizers. A frequency synthesizer generates a range of output frequencies from a single stable reference frequency of a crystal oscillator. Many applications in communication require a range of frequencies or a multiplication of a periodic signal For example, in most of the FM radios, a phase-locked loop frequency synthesizer technique is used to generate 101 different frequencies. Also most of the wireless transceiver designs employ a frequency synthesizer to generate highly accurate frequencies, varying in precise steps, such as from 600 MHz to 800 MHz in steps of 200 KHz. Frequency Synthesizer are also widely used in signal generators and in instrumentation systems, such as spectrum analyzers and modulation analyzers.

The concept of frequency synthesis is not new, it was in existence even in 1930's. But the cost of implementing a frequency synthesizer was so high that it was almost impractical for many designs. Later on when Integrated circuit technology started offering frequency synthesizer using PLL technique in a single low cost, low power chips this techniques became widely adopted for many designs.

A basic configuration of a frequency synthesizer is shown in Figure 2.1. Besides a PLL it also includes a very stable crystal oscillator with a divide by N–programmable

divider in the feedback loop. The programmable divider divides the output of the VCO by N and locks to the reference frequency generated by a crystal oscillator.

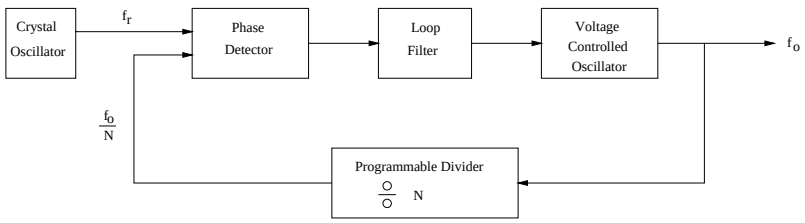

Figure 2.1: A basic Frequency Synthesizer

The output frequency of VCO is a function of the control voltage generated by the PD. The Output of the phase comparator, which is proportional to the phase difference between the signals applied at its two inputs, control the frequency of the VCO. So the phase comparator input from the VCO through the programmable divider remains in phase with the reference input of crystal oscillator. The VCO frequency is thus maintained at Nf_r. This relation can be expressed as

$$f_r = \frac{f_0}{N} \tag{2.1}$$

This implies that the output frequency is equal to

$$f_o = Nf_r \tag{2.2}$$

Using this technique one can produce a number of frequencies separated by f_r and a multiple of N. For example if the input frequency is 24KHz and the N is selected to be 32 (a single integer) then the output frequency will be 0.768 MHz. In the same way, if N is a range of numbers the output frequencies will be in the proportional range. This basic technique can be used to develop a frequency synthesizer from a

single reference frequency. This is the most basic form of a frequency synthesizer using phase locked loop technique. Its stability is dependent on the stability of the reference signal f_r.

2.1 A Design Problem

To gain some design experience and some more insight, consider a real design problem of a frequency synthesizer with the following specifications

DESIGN PROBLEM	
PARAMETER	**SPECIFICATION**
Frequency Range	924 MHz - 927 MHz
Step Size	300 KHz
Voltage Range	0.3 - 3.3 Volts

Table 2.1: A high speed frequency synthesizer

Let us call this synthesizer as the MYDESIGN synthesizer. The design of a frequency synthesizer using the principles described in the previous section is not a simple process, as it involves the design of various subsystems including the Voltage Controlled Oscillator, Phase Detector, Low–pass filter and in the feedback path, the programmable divider. The process begin from design requirements, therefore some of the design issues should be considered in the very early stages of the process. As some of the design parameters, which are related to the performance of the PLL are

dependent on the architecture of each subsystems. Such as, the effect of N on the PLL bandwidth, the limiting factors for the range of N, the upper limit of the setup time of the PLL; if N is changed, how the effect of noise will effect the purity of the output signals and the speed limit of the circuit etc. Since the addition of noise to a PLL model makes it a non-linear system, which is out of the scope of this work, thus noise consideration will not be covered here.

For illustration purposes, consider the speed limit of the circuit. One of the biggest constraint in a high speed frequency synthesizer design is the speed limit of the programmable divider N. A single divide by N unit can handle only up to 25 MHz of frequency. Therefore some special design techniques are necessary to implement a programmable divider in high speed designs. However there are many ways for overcoming this limitation of frequency. Such as

- VCO output may be fixed with the output of a crystal oscillator and the resulting remaining or the difference frequency can be fed to the programmable divider
- the VCO output may be multiplied from a low value in the operating range of the programmable divider to the required high output frequency
- Or a fixed ratio divider capable of operating at a high frequency may be interposed between the VCO and the programmable divider

All the methods discussed above have their limitations, although all have been used in many applications [3]. The first method is most useful than the other two as it allows narrow channel spacing or high reference frequencies, but it has a drawback. Since the crystal oscillator and mixer are within the loop, any crystal oscillator noise or

mixer noise appears in the synthesizer output [3]. The reamining two techniques are not as useful either. However an improved divider technique, known as two–modulus prescaling dividers exits, which is shown in Figure 2.2

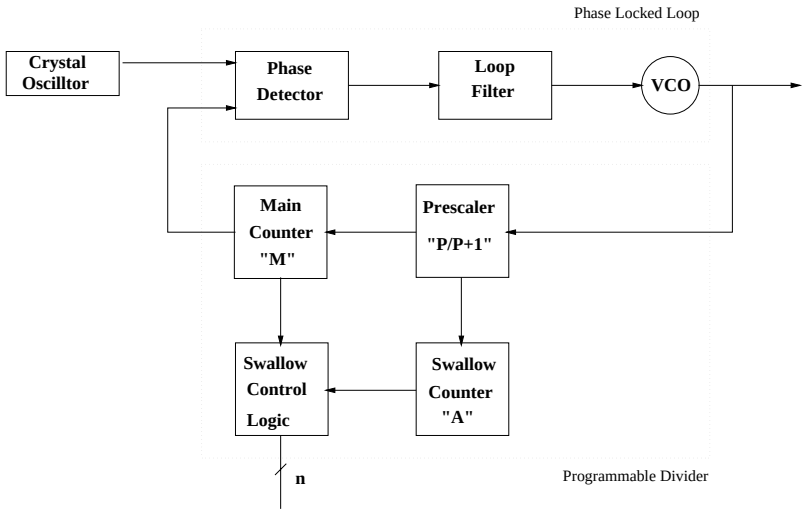

Figure 2.2: A two–modulus Frequency Synthesizer

This circuit design make use of a high frequency divider using some fixed value prescalers. In one mode it divides by P and in the other mode it divides by $P + 1$ prescalers, depending on the logic state of the control input. The prescalers reduce the high frequency by division to a lower frequency, so the rest of the circuit sees only a fraction of the high output frequency. For example, if a prescaler of 20 is used at the output of 900 Mhz, then the rest of the circuit only sees 45 Mhz. These prescalers usually can be made to handle high frequencies in the digits of MHz. In the Figure 2.2 a special low frequency counter is used to control the division ratio of the prescaler and consists of two programmable counters the swallow counter A, Main counter M and some control logic. Initially the two counters are loaded with the values M and

A, where $A \leq M$, and the modulus control signal is low, so the prescaler divides by $(P+1)$. The counters are both decremented on every rising edge of the output of the prescaler unit, until the counter A reaches zero. When A becomes zero, the modulus control signal becomes high and the prescaler start dividing by P until the value of the M counter reaches zero. At this point both the counters are reset and the process begins again. The prescaler thus divides by $(P+1)$ for the count value of the counter A and by (P) for $M-A$ times. This relation can be best explained with the following equation

$$N = A(P+1) + (M-A)P = MP + A$$

There are some limitations imposed by the architecture of the system on the values of M and A, since the two modulus presacler does not change modulus until counter A reaches zero,therefore count value in counter M should never be less than the value in counter A. This constraint also limits the minimum count, a system may reach to, which is equal to $A(M+1)$, since A is the maximum possible value, the swallow counter can have. So by varying the value of A, a large range of integer values can be obtained and so is the output frequencies. The use of this system entirely overcomes the problems of programmable divider in high speed designs. This model is also the proposed model for the MYDESIGN problem presented previously. All the counters and the control logic unit will be modeled in chapter **??**.

2.1.1 An Example

It will be easy to understand the concepts developed in section 2.1.1 with a reference example. Consider the following case where the output frequency is 115 MHz and the reference input is 5 MHz. The N is thus required to be 23 and the values of M,

A and P are selected to be as 5, 3 and 4 respectively. The method of determining the counter values will be explained in the next section.

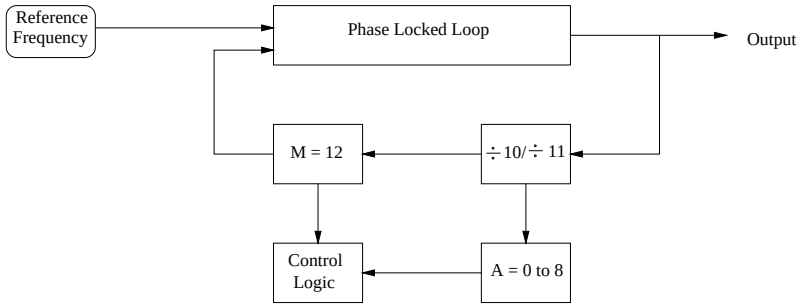

Figure 2.3: A basic Frequency Synthesizer

Initially the counters M and A will be loaded with the values 5 and 3 respectively. The value of each of these counters will be decremented after each rising pulse of the prescaler output. The more detailed timing consideration can be understood with reference to the example Figure 2.4 The prescaler divides by 5, when the A counter has the values 3,2,1 and by 4, when A becomes zero. Thus for the values 2 and 1 of the M counter, it divides by 4. The counter M outputs a pulse to the phase detector, when it becomes zero. Both the counters are reset(when M reaches zero) and the process begins again. So the whole division is 23 as is expected. The output signal is derived from the short pulse used to reset the counters. It is important to note that the presacler division ratio is determined by the state of the modulus control on the rising input edge when the prescaler output is about to become high.

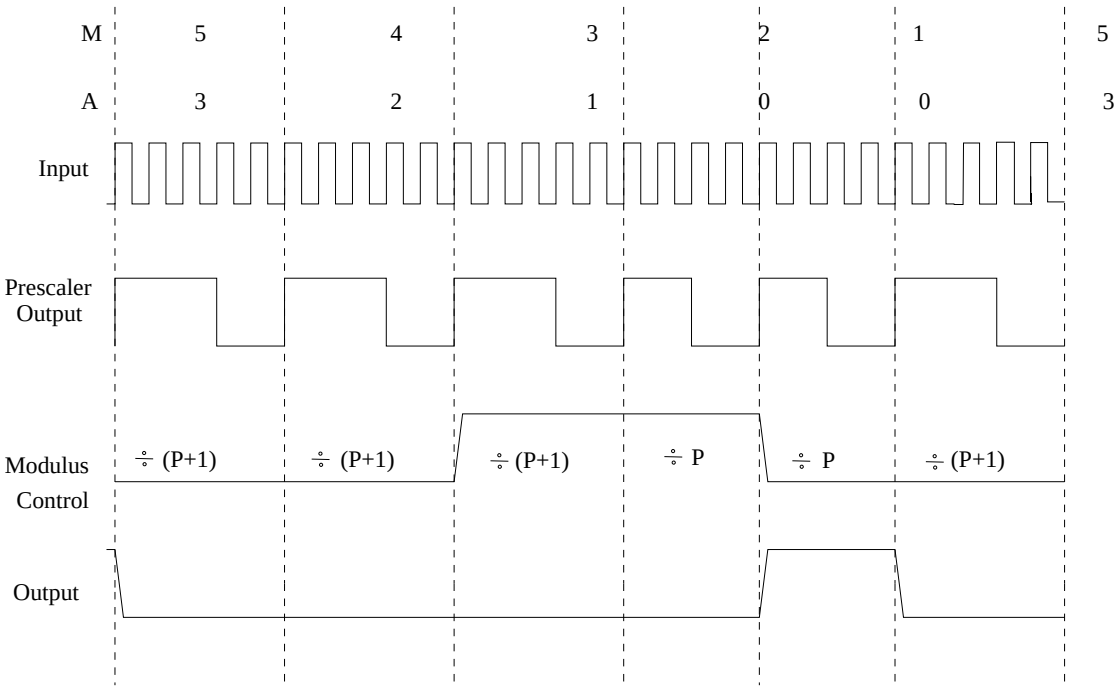

Figure 2.4: Timing Diagram of a Two-Modulus Prescaler; $N = 5, A = 5, P = 5$

2.2 Design Equations for Two–Modulus Divider

As was mentioned earlier, that there are many constraints for the values of the swallow counter A, main counter M and on the values of prescaler required for a particular design. Let us calculate these numbers for the MYDESIGN synthesizer.

$$R = Reference\ frequency = Step\ size\ of\ the\ output\ frequencies \tag{2.3}$$

Therefore, from the data provided about the MYDESIGN frequency synthesizer, we have

$$R = 300KHz \tag{2.4}$$

Then the value of N the total division number can be determined as following

$$N = \left[\frac{Output\ Frequency}{Channel\ Spacing}\right] = \left[\frac{924MHz - 927MHz}{300KHz}\right] = 3080 - 3090 \tag{2.5}$$

The value of prescaler can be selected any value, such as 8/9 or 32/33 or 64/65 etc. For instant if 64/65 is selected for the MYDESIGN design then the values of M and A can be calculated using the following equations

$$M = truncate\left[\frac{N}{P}\right] = \left[\frac{3080 - 3090}{64}\right] = 48 \tag{2.6}$$

We see that the value of M is fixed for the entire range of frequencies, that was the main reason behind selecting the prescaler values 64/65. The fix value of M also reduces the complexity of the divider. If possible, one should always try to fix the value of M. The value of A is calculated with the following equation

$$A = N - [M \times P] = N - [48 \times 64] = 8 \; to \; 18 \tag{2.7}$$

So, if the value of A is 8 the output frequency will be 924 MHz and if it is 18 then the output frequency will be 924 MHz. Thus this range of A from 8 to 18 is for the desired output frequencies, but the design is easily expandable for other frequencies outside this range. Since we have 5 output line to program the the A counter, therefor a minimum of 0 and a maximum number 31 can be loaded in this counter. By doing this the value of A is still less than the value of M, which meets with the constraint, thus all these numbers are valid. Using the above equations we see that, a 0 value of the A counter will provide us with the 921.6 MHz output frequency and the value of 31 will generate an output frequency of 930.9 MHz. All the possible intermediate frequencies are listed in Table 2.2 on the next page. From the value of A, we can also estimate the number of output lines n, required to program the swallow counter A. Since in this case the maximum value of A is 18, we need 5 output lines for interfacing and programming the swallow counter. The design of all of theses sub-functional units will be covered in the next chapter.

Swallow Counter	Divide by "N"	Output Frequency
0	**3072**	**921.6 MHz**
1	**3073**	**921.9 MHz**
2	**3074**	**922.2 MHz**
3	**3075**	**922.5 MHz**
4	**3076**	**922.8 MHz**
5	**3077**	**923.1 MHz**
6	**3078**	**923.4 MHz**
7	**3079**	**923.7 MHz**
8	**3080**	**924.0 MHz**
9	**3081**	**924.3 MHz**
10	**3082**	**924.6 MHz**
11	**3083**	**924.9 MHz**
12	**3084**	**925.2 MHz**
13	**3085**	**925.5 MHz**
14	**3086**	**925.8 MHz**
15	**3087**	**926.1 MHz**
16	**3088**	**926.4 MHz**
17	**3089**	**926.7 MHz**
18	**3090**	**927.0 MHz**
19	**3091**	**927.3 MHz**
20	**3092**	**927.6 MHz**
21	**3093**	**927.9 MHz**
22	**3094**	**928.2 MHz**
23	**3095**	**928.5 MHz**
24	**3096**	**928.8 MHz**
25	**3097**	**929.1 MHz**
26	**3098**	**929.4 MHz**
27	**3099**	**929.7 MHz**
28	**30100**	**930.0 MHz**
29	**30101**	**930.3 MHz**
30	**30102**	**930.6 MHz**
31	**30103**	**930.9 MHz**

Table 2.2: The Range of Possible Output Frequencies

CHAPTER 3

DPLL: COMPONENTS AND TECHNOLOGIES

The Phase Detectors, loop filters, and VCOs are the main components in a PLL based designs. These devices may be implemented using several different forms using several different technologies. Mostly the technical requirements of an application that control the design process usually limit the selection of these components that can be used in a particular design. Some of the common methods of selecting these components based on the specifications provided for a design are discussed in this chapter.

3.1 Phase Detector

Many different types of Phase Detectors have been used in PLLs designs. Some phase detectors accepts sinusoidal inputs and operate like analog multiplier type phase detectors and other are based on switching mechanisms, and accepts digital signals. Each type of phase detectors has its advantages and disadvantages. For example multiplier type of phase detectors provide adequate performance, even when the input signal is noisy and the sequential PDs are sometimes better because they can be made to detect both the phase and frequency differences. Generally in PLL based frequency synthesizer designs, sequential logic PD's are used as the input signals are

less noisy. There are three main digital PD's that are used for frequency synthesis applications are the following

- Phase Frequency detector
- EXOR Phase Detector
- JK flip flop PD

3.1.1 Phase and Frequency Detector

The Phase/frequency detector turns out to be the best one among all of the PDs. It offers an unlimited pull-in range which guarantees DPLL acquisition even under the worst operating conditions. A schematic diagram of the phase/frequency detector is shown in Figure 3.1

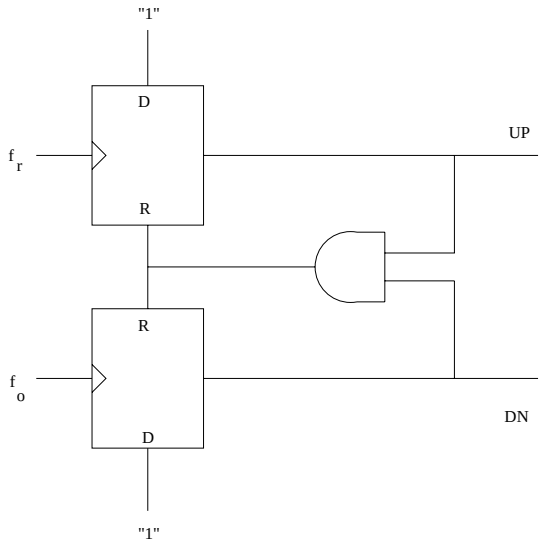

Figure 3.1: Three state Phase Detector

The operation of this circuit is based on two D-type flip flops and a simple AND gate. Each flip flop has its input wired high. Under this condition, the flip-flop with a low Q output will transition to high on the next rising edge of its clock input. Also if such an input transition occurs when Q is high, then there will be no change in the flip flop state. A high signal on a reset input will force Q low as soon as the reset signal is applied. Finally, a logical high on both of the Q outputs causes the resetting of both of the flip flops.

It generates two outputs which are not complementary of each other. The output signal depends not only on the phase error θ_e, but also on the frequency error $\Delta\omega = \omega_A - \omega_B$. If the frequency ω_A of the input A is less than the frequency ω_B of input B, then the PD generates positive pulses at the output Q_A, while Q_B remains at zero. Same is true for the other case, when $\omega_A \leq \omega_B$ positive pulses appear at Q_B and Q_A remains at zero. The width of the pulses is equal to the phase difference between the two inputs as shown in Figure 3.3. If $\omega_A = \omega_B$, then no pulse appear at either Q_A or Q_B. Thus the average value of the $Q_A - Q_B$ is proportional to the frequency or the phase difference between the inputs at A and B. The output Q_A and Q_B are usually called as the UP and DOWN signals. Depending on the operation describe above the PFD can be in one of the four states:

- UP = 0, DN = 0
- UP = 0, DN = 1
- UP = 1, DN = 0
- UP = 1, DN = 1

The fourth state is prevented, by adding an additional NAND gate in the circuit. So the circuit remains in the remaining three states only. Let us assign the numbers to various states as follows:

- UP = 0, DN = 0 —— state 2
- UP = 0, DN = 1 —— state 0
- UP = 1, DN = 0 —— state 1

To avoid dependence of the output upon the duty cycle of the inputs, the circuit should be an edge-trigerred sequential machine. Such that the circuit will change the states only on the rising edge of the transitions at the inputs A and B. A state diagram summarizing the operation is shown in the Figure 3.2

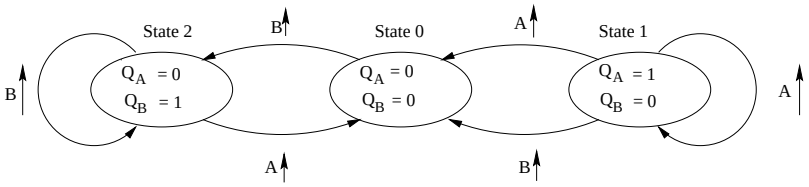

Figure 3.2: PFD State Diagram

The state of the PFD is determined by the positive edge transitions on the inputs A and B as shown in the state diagram. If the PFD is in state 0, then a transition on A will take the circuit to state 1, where The state values $Q_A = 1$ and $Q_B = 0$. The circuit remains in this state until a positive transisition occur at the input B, and the PFD returns to state 0. The transition from zero state to state2 is same as transition from 0 to 1 state. The only difference is that a a positive transition at B occurs instead of at A.

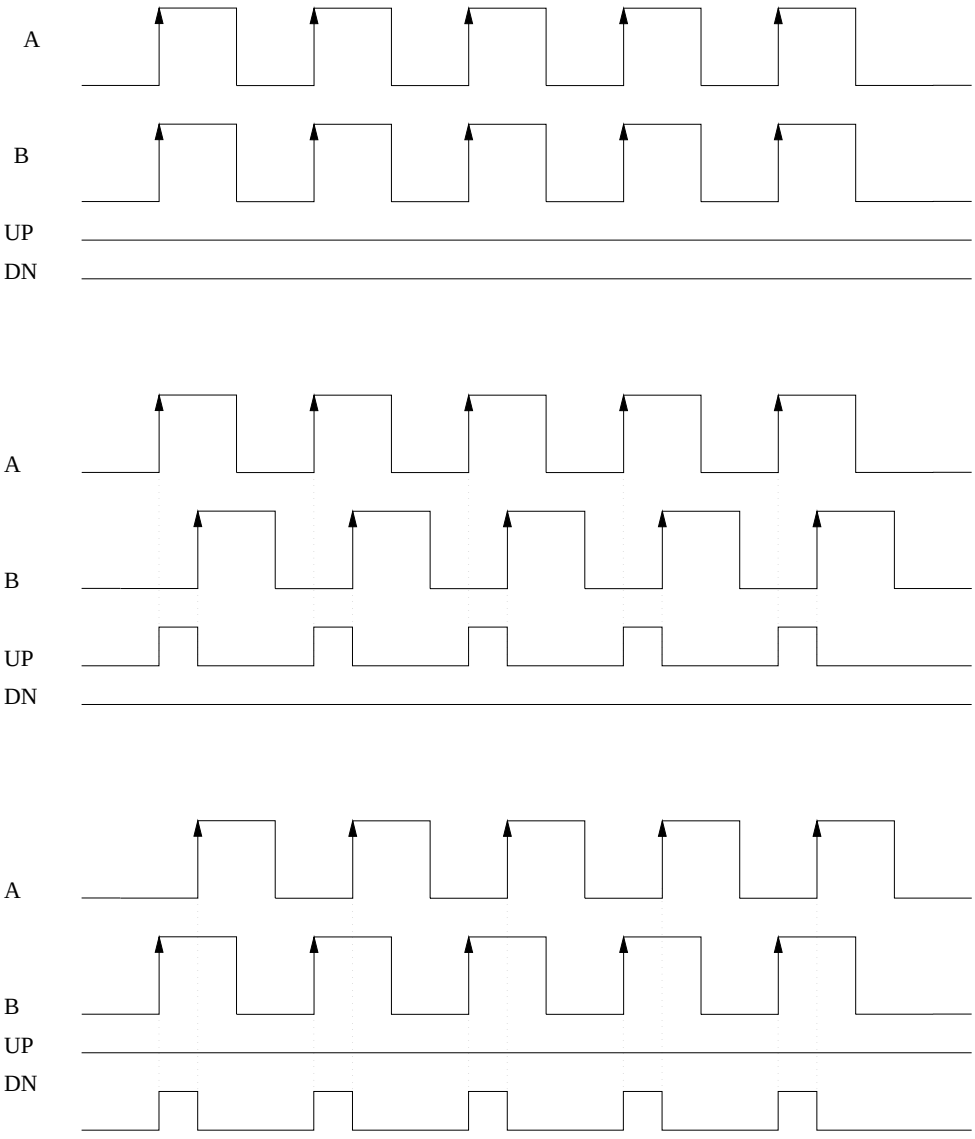

Figure 3.3: The outputs of PFD)

Based on this description of the circuit, it is easy to see that Figure 3.1 is a three state logic device. The state where both Q_A and Q_B are high is not stable and is not included in Figure 3.2, since it generates a signal that reset both flip flops.

In Figure 3.1, outputs Q_A and Q_B are the UP and DN outputs respectively. The reason behind this nomenclature comes from how these outputs are used in most applications; often Q_A and Q_B are used to drive a circuit similar to that depicted in Figure 3.4.

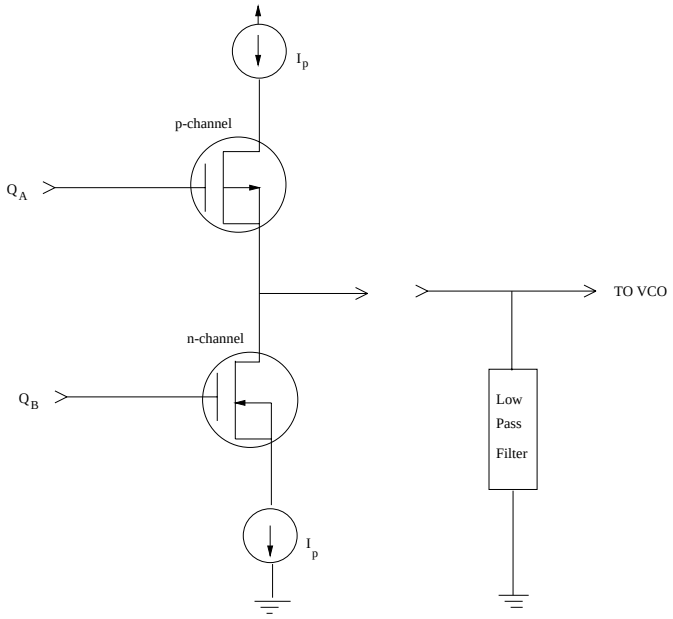

Figure 3.4: Output circuitry for use with phase/frequency detector

In this Figure each field effect transistor (FET) acts as a simple switch that closes when its input goes high. Hence terminal common to both FETs goes high when Q_A goes high, and it is grounded when Q_B goes high. In most application a high Q_A causes the loop filter to integrate some current I_p. This generates a VCO control

voltage that slew the oscillation in the proper direction. Because of this operation the circuit depicted in Figure 3.4 is a part of what is called a charge pump. It will be studied in section 3.3.

The signal I_d is thus a logical function of the PFD state. When PFD is in state 1, I_d must be positive, and when PFD is in state 2, I_d must be negative. For state o, the I_d will be zero. Theoretically I_d is a ternary signal [1]. If we plot the average I_d signal vs. phase error θ_e we get a sawtooth function as shown in Figure 3.5

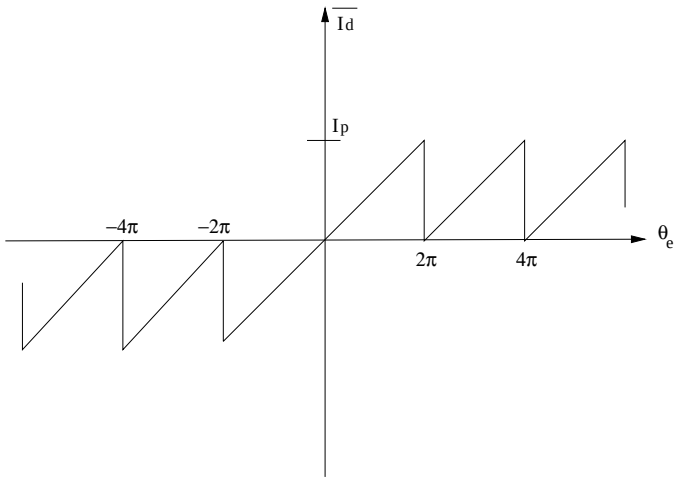

Figure 3.5: Plot of the average PFD output signalI_d vs. phase error θ_e)

The curve is linear between -2π to 2π, and then repeats every 2π. If the phase error θ_e exceeds 2π, the PFD behaves as if the phase error is rotated back to zero. Hence it is a periodic curve with a period of 2π. From Figure 3.5 the gain of PFD can be calculated and is given below

$$K_d = \frac{I_p}{2\pi} \tag{3.1}$$

Furthermore, if the input frequency ω_i is greater than that of the output frequency ω_o, it implies that at input A more transitions occur as compared to input B. In this situation the PFD output states will toggles only between states o and 1, but will never go into state 2. If $\omega_i >> \omega_o$, then PFD will remain in state 1 most of the time. When $\omega_i < \omega o$, PFD toggles between state state 2 and 0, and if $\omega_i << \omega_o$ then it will remain in state 2 most of the time. Therefore we can conclude that the average output signalv_d of PFD varies monotonically with frequency error $\Delta\omega = \omega_i - \omega_o$, when DPLL is in tracking or out-of-loack mode. Since it can detect both phase and frequency errors between the input signal it is best one to use in frequency synthesis application.

3.1.2 PFD Performance

The simulink toolbox of MATLAB was used to simulate the PFD model. The model is sh

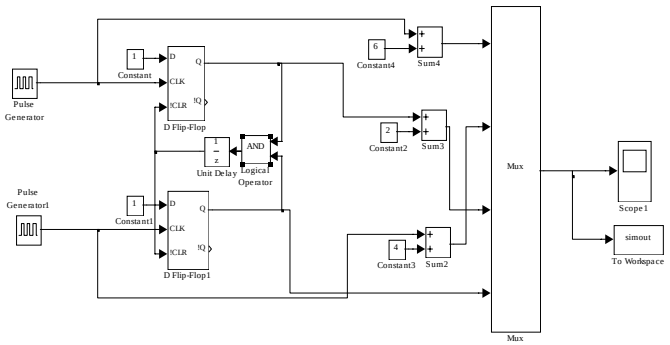

Figure 3.6: PFD's Simulink Model

The performance of PFD is as expected. In Figure 3.7, the frequencies at input A and B of PFD are same, but A leads B, and therefore the pulses appear at output

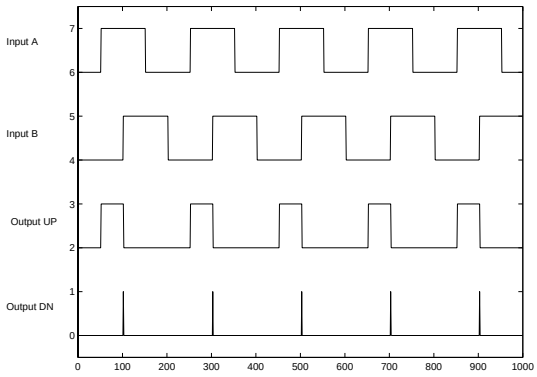

Figure 3.7: PFD SIMULINK Results

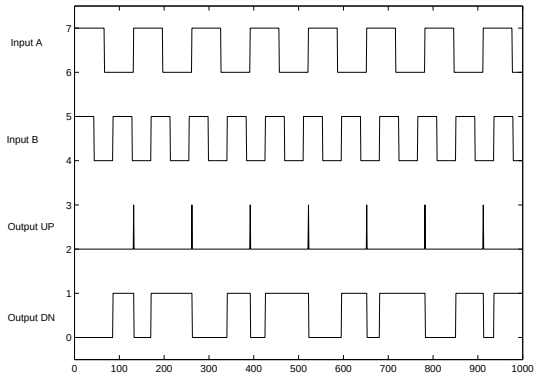

Figure 3.8: PFD simulation

UP. In Figure 3.8 input B leads A and the frequency at input B greater than A, and the pulses appear at output DN and the widths are coressponding to the difference between the tow inputs. Thus the designed PFD is operating as designed.

3.2 Charge Pump

The output of a PFD can be converted to DC (voltage/current) in many different ways. One approach is to sense the difference between the two outputs by using a differential amplifier and apply the result to a low pass filter [5]. The second method is by using a charge pump.

A charge pump is three state design. It takes two inputs out from the PFD and outputs a DC current or voltage. The charge pump consists of two current sources and the output of the charge pump drives the low pass filter. The charge pump either charges or discharges a capacitor with voltage or current pulses. A filter is used to limit the rate of change of the capacitor voltage, and the result is a slowly rising or falling voltage that depends on the frequency difference between the PLL output voltage and the reference frequency. The VCO increases or decreases its frequency of operation as the control voltage is increased or decreased [7]

The designed charge pump design is shown in Figure 3.9 It feeds pump current I_p pulses to a filter, whenever the output of the PFD is in state 1 or state 2. For state 0 of PFD it acts as an open circuit for the loop filter. The current I_p results in charging or discharging the capacitor voltage. The polarity of the charging current I_p is positive, if pulses appear at UP output of the PFD, and is negative when pulses appear at DOWN output of the PFD.

In Figure refpump, the SW1 is the switch between state 1 and 0. When pulses appear at UP output of the PFD, the switch is closed and the pulses of charge current charge up the loop filter capacitor. The switch SW2 is a switch between state 2 and 0. When pulses appear at DN output of PFD, the switch is closed and the current from the loop filter flows down to ground, thus discharging the capacitor. So the

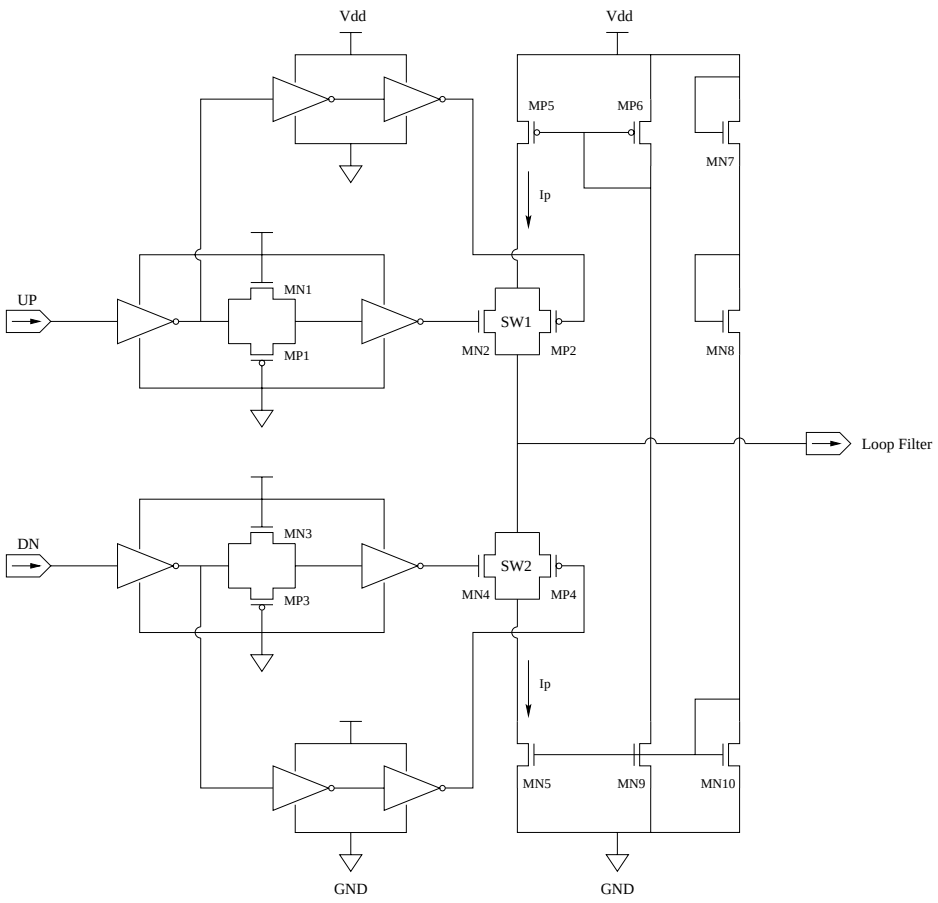

Figure 3.9: Designed Charge Pump for MYDESIGN

voltage at the loop filter capacitor rises and falls, and control the VCO. The VCO in response either increase or decrease its frequency.

The designed charge pump also consists of current sources namely MN7, MN8, and MN10 with n(MN9) and p(MP6) current mirror sources, which feed current mirror transistors MP5 and MN5. These are connected to loop filter via the switches (SW1, SW2), with complementry clocks balanced for equal time dealy. The current-source transistors are double the minimum length to improve the drain conductance. The average current $I_d = I_p$ can be calculated from the following Figure 3.10

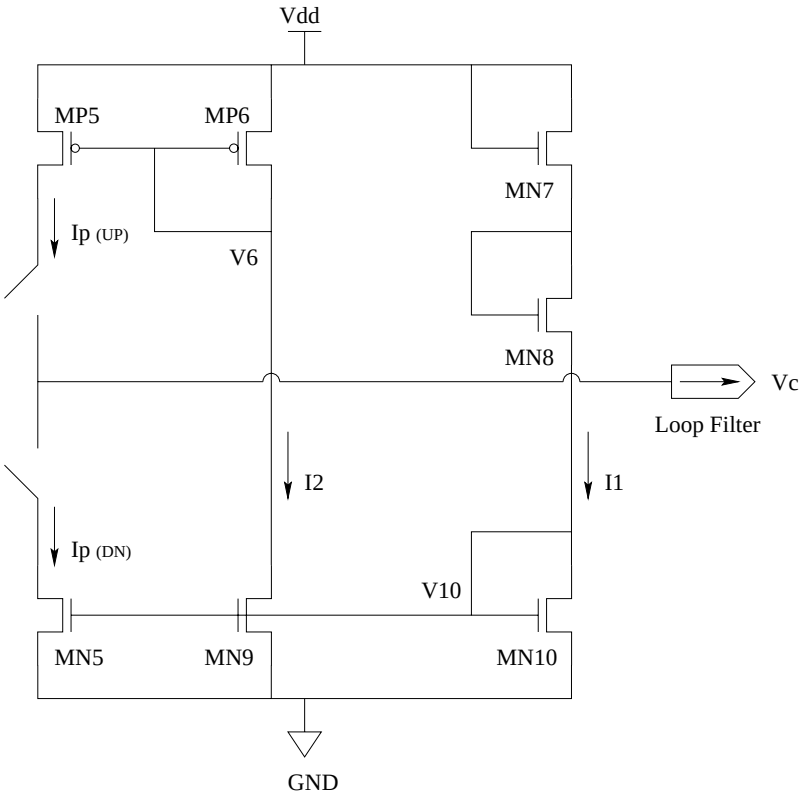

Figure 3.10: Current Mirror in Charge Pump

The used equation are listed below

$$K_{n(p)} = \frac{1}{2}\mu_{n(p)} C_{ox} \frac{W}{L} \tag{3.2}$$

$$V_{10} = \frac{V_{dd} - 3V_{tn}}{1 + \sqrt{\frac{K_{n10}}{K_{n7}}} + V_{tn}} \tag{3.3}$$

$$V_6 = V_{dd} + V_{tp} + \sqrt{\frac{K_{n9}}{K_{p6}}(V_{10} - V_{tn}} \tag{3.4}$$

$$I_1 = K_{n10}(V_{10} - V_{tn})^2(1 + \lambda V_{10}) \tag{3.5}$$

$$I_2 = \frac{K_{n9}(1 + \lambda V_6}{K_{n10}(1 + \lambda V_{10}} I_1 \tag{3.6}$$

$$I_{p(DN)} = \frac{K_{n5}(1 + \lambda V_c}{K_{n10}(1 + \lambda V_{10}} I_1 \tag{3.7}$$

$$I_{p(UP)} = \frac{K_{p5}(1 + \lambda(V_c - V_{dd}))}{K_{p6}(1 + \lambda(V_6 - V_{dd}))} I_2 \tag{3.8}$$

For $I_{p(UP)} = I_{p(DN)}$, we have

$$\frac{K_{n5}(1 + \lambda V_c)}{K_{n9}(1 + \lambda V_6)} = \frac{K_{p5}(1 + \lambda(V_c - V_{dd}))}{K_{p6}(1 + \lambda(V_6 - V_{dd}))} \tag{3.9}$$

For λ is small, we have

$$\frac{K_{n5}}{K_{n9}} = \frac{K_{p5}}{K_{p6}} \tag{3.10}$$

The designed CMOS model of the charge pump is shown in FIg. 3.11, with discharging/charging current $I_p = 10\mu A$.

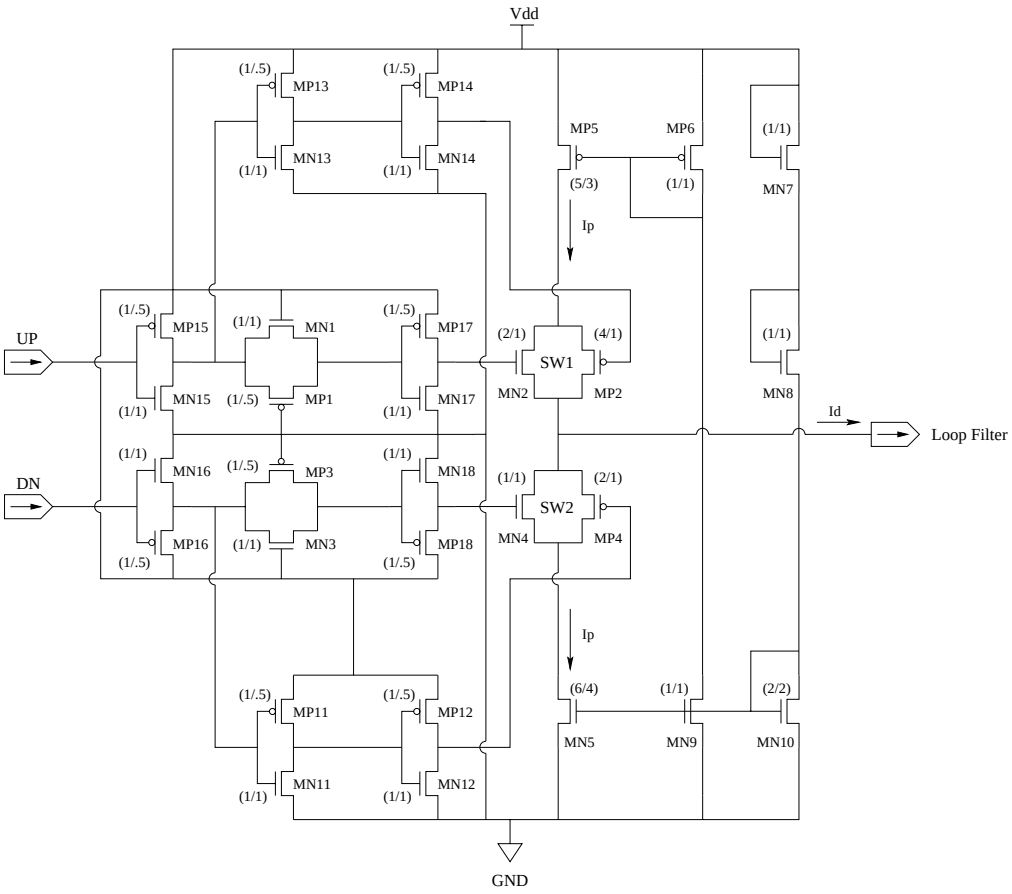

Figure 3.11: Charge Pump for MYDESIGN

3.3 Loop Filter

Both active and passive loop filters can be used in the PLL design. Most modern application utilize an active filter based on high performance operational amplifier technology. Passive loop filters are simple to design and will thus be considered for the MYDESIGN.

The design of the loop filter is the principle tool in selecting the bandwidth of the PLL. A PLL without a loop filter result in a first order system. First order system are rarely used as they offer little noise suppression. Since higher order loop filters offer better noise cancelation, loop filters of order 2 and more are used in critical applications, such as in the case of frequency synthesizer. The designed loop filter for the MYDESIGN is shown in the Figure 3.12

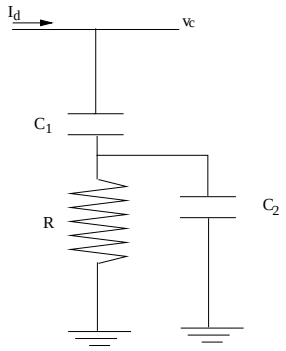

Figure 3.12: Second Order Passive Loop Filter

It is a second order loop filter, as it contains two capacitors in the circuit. The transfer function of the loop filter is

$$Z(s) = Z_h \frac{s + \omega_2}{s(\frac{s}{\omega_3} + 1)} \tag{3.11}$$

where

$$Z_h = \frac{R(C1 + C2)}{C1} \tag{3.12}$$

$$\omega_2 = \frac{1}{R(C1 + C2)} \tag{3.13}$$

$$\omega_3 = \frac{1}{RC2} \tag{3.14}$$

Selecting $\omega_2 = K/4$ and $\omega_3 = 4K$, we have the following relation for resistor and capacitors of the loop filter

$$R = \frac{15\pi NK}{8I_pK_o} \tag{3.15}$$

$$C1 = \frac{2I_pK_o}{\pi NK^2} \tag{3.16}$$

$$C2 = \frac{2I_pK_o}{15\pi NK^2} \tag{3.17}$$

3.4 Chrage Pump and Loop Filter Performance

Charge Pump is circuit that converts two digital signals of PFD to one output, which is currentI_p in the MYDESIGN charge pump. It was simulated using SPICE and the results are shown in Figure 3.13. The first plot consists of the two inputs to the charge pump and the I_p current at the output node of the chrage pump. It is clear from the plot that charge pump deliver a current $I_p = 10uA$ to the filter for the

input UP of the PFD and discharges the loop filter for input DN of the PFD. The intensity of the current is same in both directions. The second plot is the UP and DN inputs and the resulting control voltage v_c to the VCO. We see that the control voltage increases in response to the differences in the UP and DN inputs. Therefore the circuit is working properly.

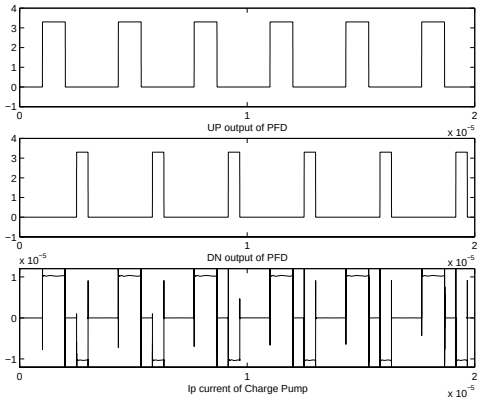

Figure 3.13: Spice Simulation of Charge Pump

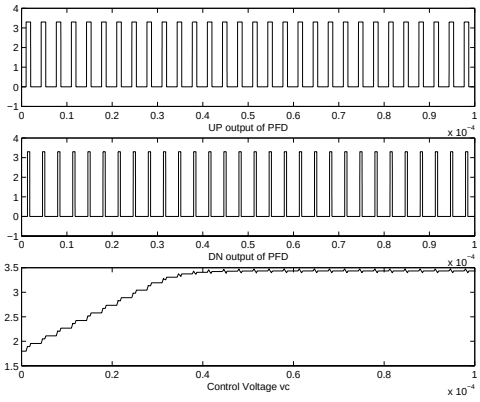

Figure 3.14: Spice simulation result for Control Voltage

3.5 VCO

A VCO can be realized using a wide rage of technologies in many different ways [6], but there are two basic general classes. One is relaxation oscillators(or astable multivibrators) and the other is resonant oscillators(or Vanderpole oscillators) [7]. The main difference in these two classes is digital and analog outputs. Since MYDESIGN is a DPLL circuit we will look at relaxation type VCO designs.

For MYDESIGN synthesizer the operation range of the VCO can be found from the specification provided. The operating range of the VCO is equivalent to the band of the output frequencies; which in this design is $924MHz - 927MHz$. Since the voltage range is restricted to $0.3Volts - 3.3Volts$, the VCO gain can be determined with the aid of Figure gain

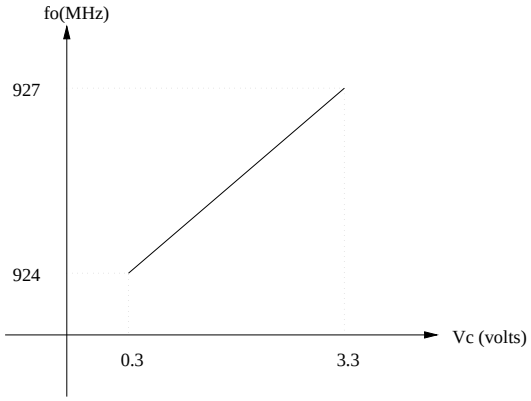

Figure 3.15: VCO gain

So the VCO gain is

$$VCOgain = K_o = \frac{(927 - 924)MHz}{(3.3 - 0.3)Volts \times 2\pi} = \frac{1}{2\pi} MHz/Volt \tag{3.18}$$

Different circuit configurations and technologies offer different capabilities and performances. As mentioned previously that the selection of a circuit configuration and technology is driven by the requirements of an application, and the selection process usually involves tradeoffs and compromises [6].

CHAPTER 4

PLL SIMULATIONS AND RESULTS

Frequency Synthesizer using Phase Locked loop is a feedback Control System. In such a system, the time domain performance specification are important indices because control systems are inherently time domain systems. It is necessary to determine initially whether the system is stable. If the system is stable then the response of the system to a specific input signal will provide several measures of the performance. Normally a standard test input signal is chosen for the response of the system. This approach is usually quite useful, because there is direct correlation between the response of a system to a standard test input and the system's ability to perform under normal operating conditions. The standard test input signals commonly used are the step input, the ramp input. The ramp signal is the integral of the step input. Since step input signal is easiest to generate and evaluate and is usually chosen for performance test [2]. In this chapter first we will look at if the designed PLL is stable or not, then we will look at its time domain performance by using a step input as a test signal. We will also look at the frequency domain analysis of the designed PLL.

4.1 PLL Bandwidth

The transfer function of the PLL, based on the proposed sub-function units in chapter 3, is

$$H(s) = \frac{G(s)}{1+G(s)} = \frac{\theta_r(s)}{\theta_o(s)} = \frac{Ks + K\omega_2}{\frac{S^3}{\omega_3} + s^2 + Ks + K\omega_2} \tag{4.1}$$

Where, the forward path transfer function $G(s)$ is

$$G(s) = K\frac{s+\omega_2}{S^2(\frac{s}{\omega_3}+1)} \tag{4.2}$$

K is the bandwidth of the PLL, and is equal to

$$K = K_d Z_h K_o / N \tag{4.3}$$

Where

- K_d is the gain of the PFD $= \frac{I_p}{2\pi}$ A/rad
- Z_h is the high frequency gain of the loop filter and is given by $\frac{R(C1+C2)}{C1}$
- K_o is the gain of the VCO $= 2\pi$ Mrad/sec/V

In PLL design, the incorporation of the low pass filter in the loop provide the designer, the flexibility of choosing the bandwidth of the PLL. In typical designs, loop $bandwidth = 1/10 Input\ Frequency$ guarantees stability. So for MYDESIGN frequency synthesizer the PLL bandwidth must be

$$Bandwidth\ of\ PLL \leq 30K\ Hz \tag{4.4}$$

Since a small bandwidth results in less aquisistion time, we select the

$$Bandwidth\ of\ the\ MYDESIGN\ PLL = 20K\ Hz \tag{4.5}$$

4.1.1 Component Values

The forward path loop gain of PLL is given by

$$G(s) = K_d Z_h K_o / s \tag{4.6}$$

At high frequencies, the magnitude response of $Z(s)$ is equal to Z_h. If this is true, then the magnitude of G(s) is equal to 1, for $\omega = K_d Z_h K_o$. Based on this, the bandwidth of the PLL is given by the relation

$$\omega_{3dB} = K_d Z_h K_o = K \tag{4.7}$$

This result of bandwidth has assumed that the magnitude of Z_s is equal to Z_h, but it is only possible when $\omega > \omega_2$. Therefore

$$\omega_2 < K \tag{4.8}$$

is the constraint for selecting the value of ω_2. Also if the value of $\omega_2 < 4K$, then it will have little effect on the magnitude response of the closed loop transfer function of the PLL [8]. So we can select $\omega_2 < K/4$. We will see more about this, when considering step response of the PLL.

The selection of ω_3 is based on the same approach that we want that the $\mid G(j\omega) \mid$ crosses unity at $\omega = K$, and the PLL bandwidth will be equal to K. It is true in general that if ω_3 is not too close to K, such that $\omega_3 > K$ then the bandwidth of the PLL will remain K. The step response of the PLL will be same as for a second order PLL, as shown in Figure 4.1 if we select $\omega_3 = 4K$.

The transfer function of a second order PLL in general is given by

$$H(ss) = \frac{2\zeta\omega_n s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{4.9}$$

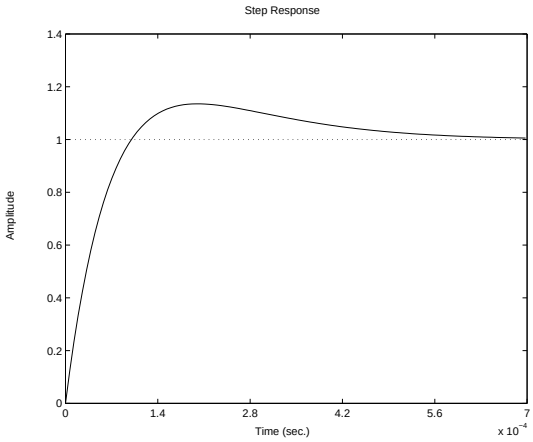

Figure 4.1: Step Response of a second order PLL $\omega_2 = K/4$

where

$$\zeta = 0.5\sqrt{K/\omega_2}$$

$$\omega_n = \sqrt{K\omega_2}$$

The result of the above discussion are re-presented in the following table 4.1

For Stability	For Optimal Performance
$\omega_2 \leqslant K$	$\omega_2 = K/4$
$\omega_3 \geqslant K$	$\omega_3 = 4K$

Table 4.1: The selected values for ω_2 and ω_3

The values of resistors and capacitors, using the formulas given in section 4.2 are found to be

K	20 K rad/sec
R	5.775 M ohms
C1	32.4675 pF
C2	2.1654 pF

Table 4.2: Calculated values of R,C1,C2 for K = 20 KHz

4.2 Loop Stability

The stability of a feedback system is related to the location of the roots of the characteristic equation of the system transfer function in the s plane. We will look at few different methods to determine, if the designed PLL is stable.

4.2.1 Root Locus of Third order PLL

A system is stable if all the poles of the transfer function have negative real parts. We can show that all the poles poles of the transfer function $H(s)$ of designed PLL, are all in the left s plane, therefore the designed system is stable. However, it is also very important to determine, how the roots of the chracterstic transfer function move around in the s plane, as we change one of the parameter of the system. Usually this parameter is selected to be the gain, the bandwidth of the PLL. One method that depicts this behavior of the roots; as gain of the transfer function is increased is Root Locus. A root locus plot of the transfer function of the designed PLL is shown in Figure 4.2

We see as K is varied from 0 to infinity, the roots remain in the left half plane, so the system remain stable, the value of K, where the two poles split and go towards

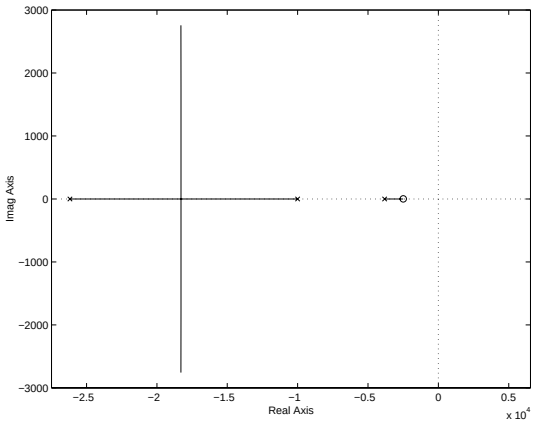

Figure 4.2: Root Locus of Third Order Designed PLL

infinity, is the point, where the system becomes oscillatory, but is still stable. Increasing K, thus will increase the oscillatory behavior of the system in time domain. We will talk about this more in the next section.

4.3 Step Response of PLL

The step response of a system is basically its time domain performance. Step response of a system provide details about settling time and percent overshoot parameters. Settling time of a step response has a direct relation to the PULL-IN-TIME parameter of the PLL. The percent overshoot provide details about system's oscillatory behavior. The relation between these two parameters is inversely proportional. We will see reducing one results in increasing the other. So the compromise is made at the best possible point to optimize the performance of the system. Figure 4.3 is the step response of the designed PLL for different values of $w2$.

As we showed previously that ω_2 should be less than K, in this plot we see how the selection of ω_2 for various different values effects the step response. It seems from

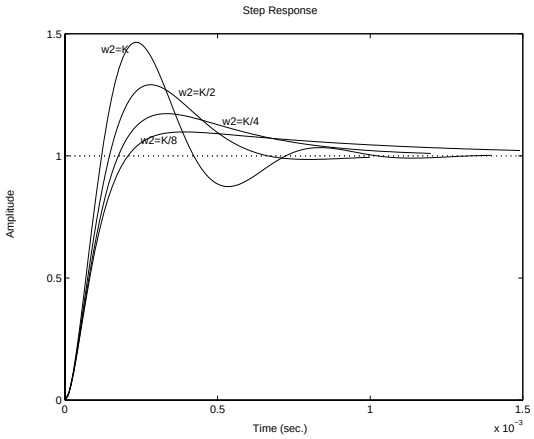

Figure 4.3: Step Response of Third Order Designed PLL

the step response plot that, it is always best to select $w2$ as small as possible. This slightly slows the response, but it make the system very stable, as overshoot is very less. However a small value of $w2$ implies large capacitor, and they take longer to charge during lock acquisition. Therefore a good compromise is to select $w2 = k/4$, this assures fast acquisition [8] and the resulting step response is shown in Figure 4.4

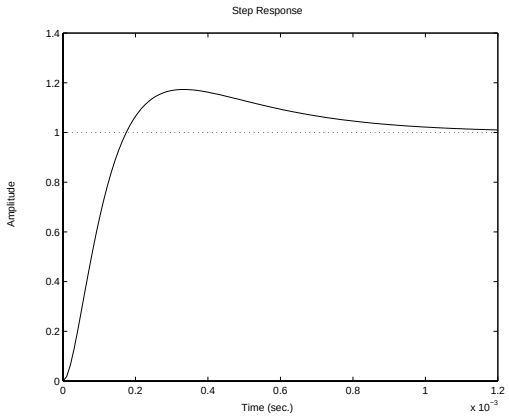

Figure 4.4: Step Response of Third Order Designed PLL

4.4 Bode Plots

The behavior of the transfer function as the frequency of the signal is varies is an important chracterstic of a circuit. One effective way to describe how the amplitude and phase angle of $\mid H(j\omega) \mid$ varies with frequency is by Bode Plots [4]. MATLAB was used to generate the bode plots for a second and a third order PLL transfer functions. The main aim was to check the bandwidth of the third order designed PLL for selected parameters and comparing it with a second order PLL. A bode plot for second order PLL is shown in Figure 4.5

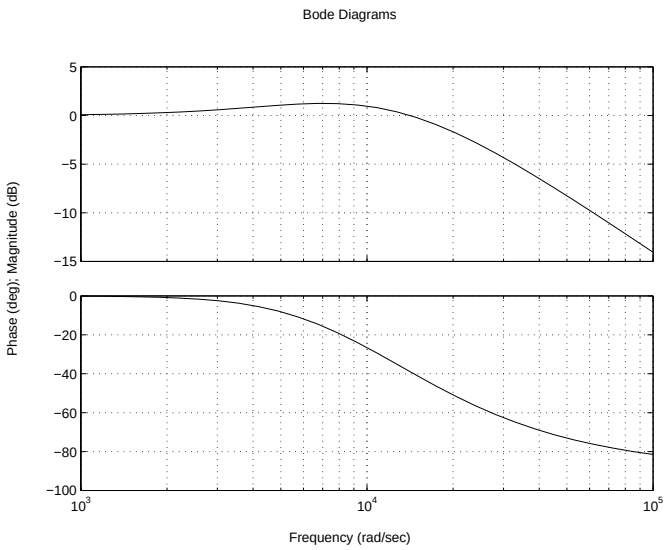

Figure 4.5: Bode Plot of second order PLL open loop, with $\omega_2 = K/4$

A bode plot for a third order designed PLL is shown in Figure 4.6 The bandwidth of the PLL is at 20 KHz. So the transfer function of the PLL is correctly working. Both plots are also very much like, thus we can conclude safely that there is no effect in the magnitude response by the third pole of the PLL.

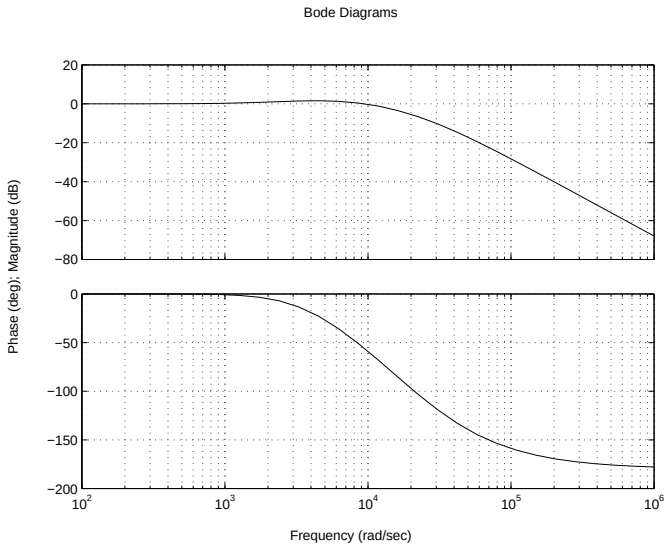

Figure 4.6: Frequency Response of Third Order Designed PLL

4.5 PLL Performance

The performance of the whole PLL model was tested using a Simulink Model shown in Figure 4.7 The parameter used for the models are as $K_o = 100MHz/v$,

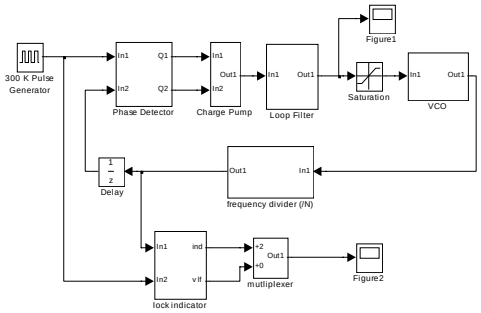

Figure 4.7: Frequency Response of Third Order Designed PLL

$K = 20KHz$, $I_p = 10uA$, $f_{free} = 925.1MHz$ and $f_0 = 927MHz$. The output of the

loop filter is shown in figure 4.8. The performance is as expected, we see the output rises, when the two inputs to the PD are out of lock and when they are in lock; it saturates. We used the lock indicator to detect when the pll is in lock. Figure 4.9 shows the lock indication and corresponding to the VCO control voltage. The Results are satisfactory.

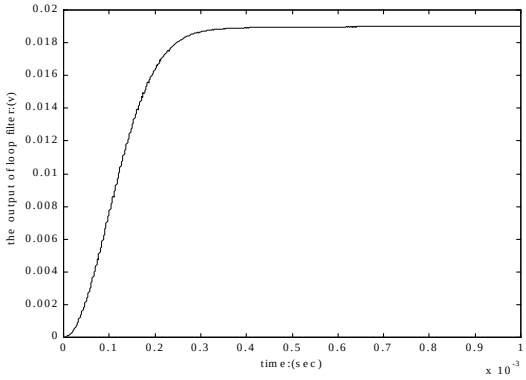

Figure 4.8: Control Voltage v_c

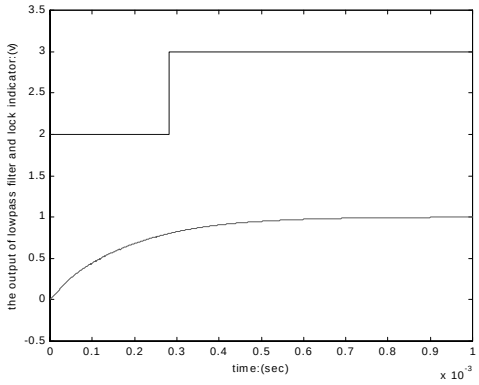

Figure 4.9: Lock Indication Curve and Control Voltage

CHAPTER 5

DESIGN OF TWO MODULUS DIVIDER AND SIMULATION RESULTS

For high speed frequency synthesizer designs incorporate high speed Dual-Modulus or Multi-Modulus dividers. Such circuit divide the input frequency by one of the moduli according to a control input [5]. A circuit diagram of a Two Modulus Divider for the MYDESIGN synthesizer is shown in Figure 2.2. It consists of two Prescalers, a Main counter, A Swallow counter and a control unit. In this chapter we will develop the gate level design for these subunits. To verify the design Matlab's SIMULINK tool box will be used.

5.1 Prescaler

Frequency dividers are also called prescalers. There are two prescalers in the two modulus divider for the MYDESIGN namely, a divide by 64 and a divide by 65 prescalers. The designed prescalers and their simulation results for the MYDESIGN are presented in the following sections.

5.1.1 Divide by 64

Asynchronous dividers are the simplest form of prescalers. They consists of a series of D flip flops, where each D flip flop's inverted output is connected back to

its input, making it a divide by two circuit. If the input is fed into the clock signal of this circuit the output frequency will be half of the input frequency. A circuit configuration of such a circuit, and its input output behavior is shown in Figure 5.1.

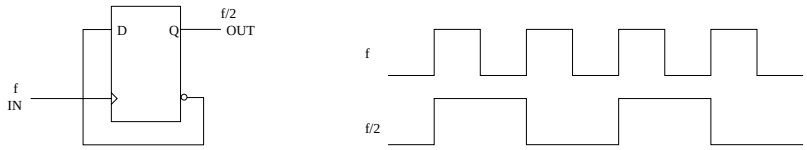

Figure 5.1: A Divide by 2 Prescaler

A very nice feature of this circuit is that the output is perfectly symmetrical square wave regardless of whether the input square wave is symmetrical or not. By cascading several D flip flops in the same configuration, it is easy to make a divide-by-2^n circuit. The non-inverting output of one flip flop can be used as an input to the next flip flop to make it a divide by 4 circuit. Thus to divide an input frequency by 64, we only need to have 6 D flip flops connected in this configuration. The designed circuit for the prescaler 64 is shown in Figure 5.2

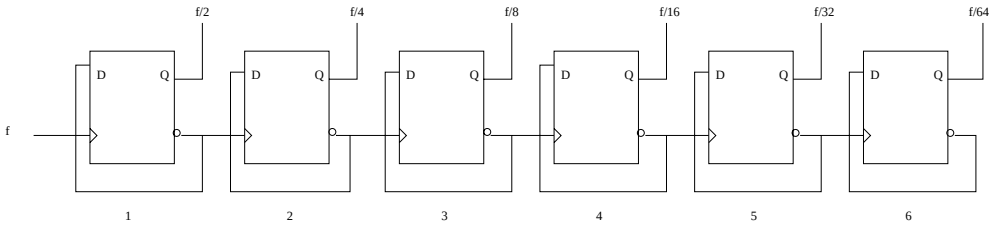

Figure 5.2: A divide by 64 Prescaler

The simulation of the prescaler 64 was performed using SIMULINK model, and the Figure 5.3 shows the input and output waveforms of the circuit.

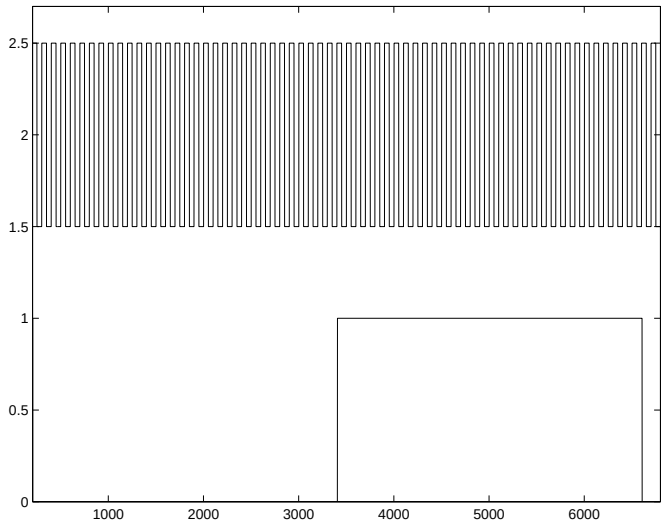

Figure 5.3: Simulink results of Divide by 64 prescaler unit

The circuit is performing as expected, one period of output square wave is equal to the 64 periods of the input square wave. Thefore output frequency is equal to the input frequency divided by the integer value 64. The duty cycle is 50 percent in this case, but it is only possible when division is not an even number. We will discuss about this more in the other divider designs.

5.1.2 Divide by 65

This prescaler is more complicated to implement as compared to a divide by 64 prescaler. The reason is the odd number division. There are two ways to build this circuit, one is completely synchronous and the other is mixed. Since the first method is more complex as compared to the second one, we will use the second method, which

is asynchronous and synchronous mixed design. In this method the circuit is divided into two units. One unit is a divide by 5 circuit and the second one is a divide by 13. The output of first unit will be fed into the second unit, and the whole circuit will be a divide by 65 prescaler circuit. These two circuits are basically ring counters with the number of states corresponding to the division number. For example if we need to divide by 5, the ring counter will have five stages only and will count in a ring fashion. Same is true for a divide by 13 unit, it will have 13 stages and will also count in the ring fashion. The following two tables shows the various stages of both ring counters.

The number of flip flops required can be found from the number of stages. So we need 3 D flip flops for the divide by 5 circuit , as it has only five stages and 4 D flip flops for the divide by 13 circuit. All the flip-flops are rising edge triggered. The designed prescaler is shown in Figure 5.4.

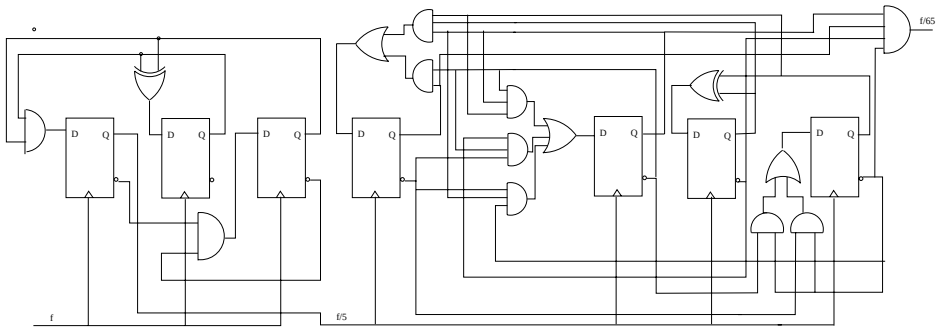

Figure 5.4: A divide by 65 Prescaler

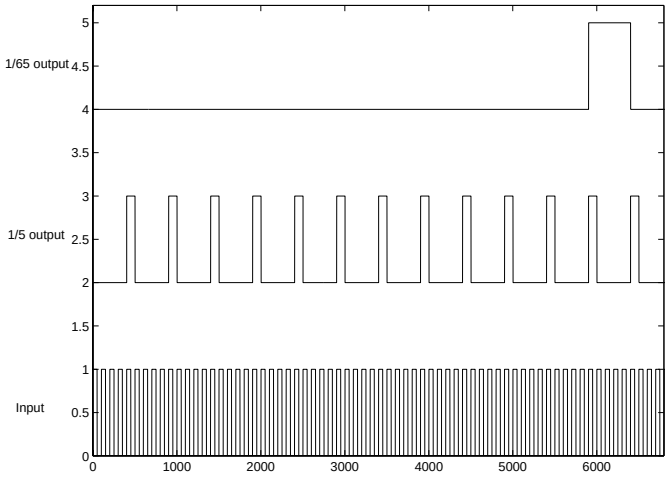

Figure 5.5: Simulink results of Divide by 65 prescaler unit

5.2 Swallow Counter

A Swallow counter in the Two Modulus Divider is a programmable down counter. For the MYDESIGN frequency synthesizer, the Swallow counter needs to count down from the loaded number to zero. It repeats the counting down sequence from the same number until the loaded number is changed externally. It has the option of Load, this is required to enable the loading into the swallow counter with the division number. To generate 10 different frequencies, the range of numbers that needs to be loaded in the swallow counter is 8 through 18. Thus 5 flip flops are required for the design. Based on the design of the swallow counter up to 32 output frequencies can be synthesized with no extra hardware. The swallow counter is reset through the LOAD value, whenever the Main counter outputs a pulse and we need to begin the counting sequence again. The proposed design of the Swallow counter for the MYDESIGN frequency synthesizer is shown in Figure 5.6

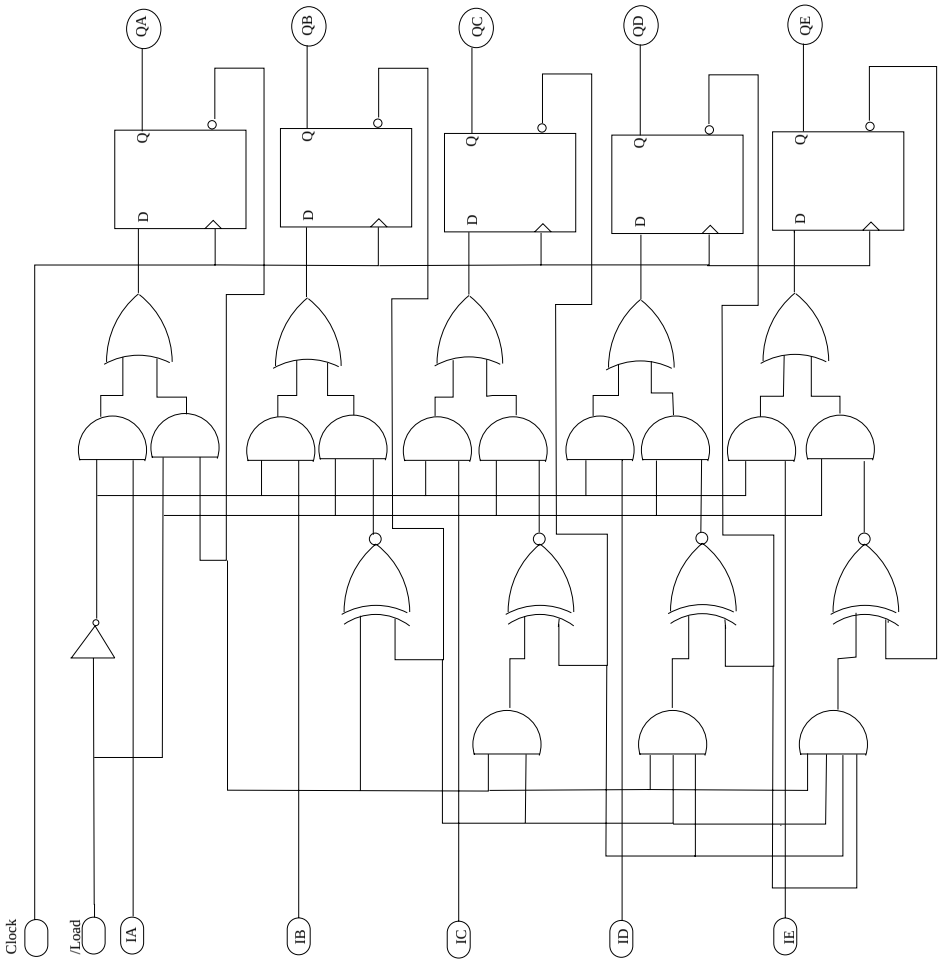

Figure 5.6: Swallow Counter

The simulation results of the swallow counter are shown in Figure 5.7. It is working as was desired.

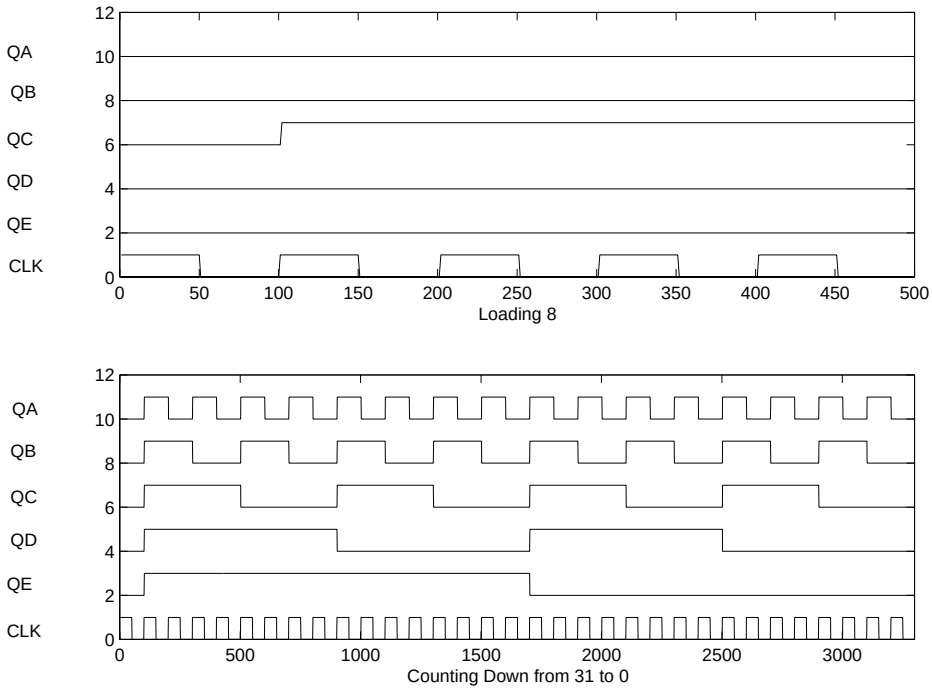

Figure 5.7: SIMULINK results for Swallow Counter

5.3 Main Counter

The main counter is also a frequency divider circuit,just like the prescalers we discussed before. It divides by 48. The number 48 is for the MYDESIGN and was derived in Chapter 2. The input of the main counter is the output of the prescaler unit. For the division number of 3080, it takes first 8 pulses of the divide by 65 prescaler and remaining 40 pulses of divide by 64 units and yield only one pulse. This pulse is is equal to the output of the VCO divided by 3080. The circuit is designed

with the same technique as we used for the divide by 65 unit. It first divide the input by 16 and then the second unit divide this output by 3 yielding a total of divide by 48 output. The circuit configuration is shown in Figure 5.8. The design was verified using SIMULINK and the resulting waveforms are shownsin Figure 5.9

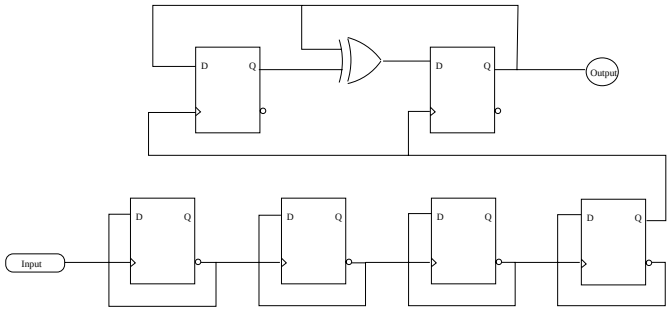

Figure 5.8: Main Counter

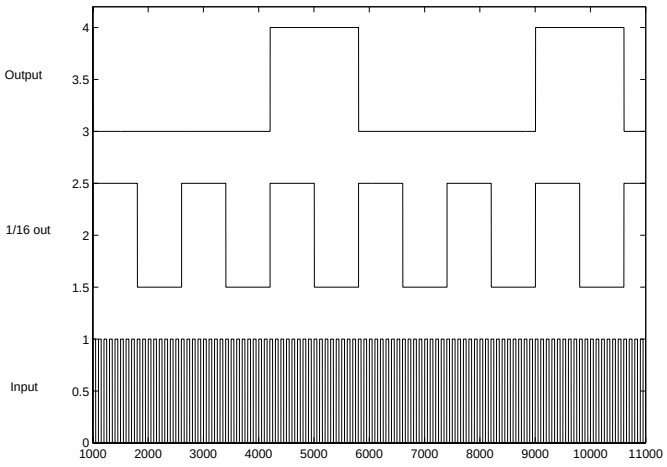

Figure 5.9: Simulink Results of Main counter

5.4 Control Logic

The designed control logic of Two Modulus Divider is shown in Figure 5.10. It consists of two (2 input) multiplexer and some other simple logic design. It will be easy to understand if, we will start from the output of the Swallow Counter. All the outputs (5 in MYDESIGN) of the Swallow Counter are fed into a OR gate. The output of the or gate is the control input of MUX2. At the very first cycle,the output of the OR gate is zero so it enables the loading operation. As soon as the Swallow counter is loaded with a value the output of the OR gate becomes high at next clock cycle, thus disabling the LOAD operation of the Swallow Counter. The high output of the OR gate, because of control input of MUX2 now enables the CLOCK input of the the swallow counter. The inverted output of the OR gate also act as a control input for the prescaler units, as is depicted in the Figure 5.10. So whenever the OR gate output is high the Divide by 65 prescaler unit is active and the input is being divided by 65;and the MAIN counter receives the pulses from the divide by 65 unit. At every pulse to the MAIN counter, the swallow counter counts down and when it reaches zero, the output of the OR gate becomes zero, and now the divide by 64 unit is active and the the CLOCK input of the swallow counter is now connected to the output of the MAIN counter again. Since at the output of the Main counter there is no pulse, because it is still receiving pulses from the divide by 64 unit and hasn't yet received 48 pulses the Swallow counter is pause during this time. When the pulses to the MAIN counter reaches 48, the MAIN counter outputs a pulse. and the Swallow counter is once again loaded with the division number and the sequence begins again.

The Control Unit was verified by simulating the whole Dual Modulus divider in the SIMULINK. The result is depicted in the Figure 5.11

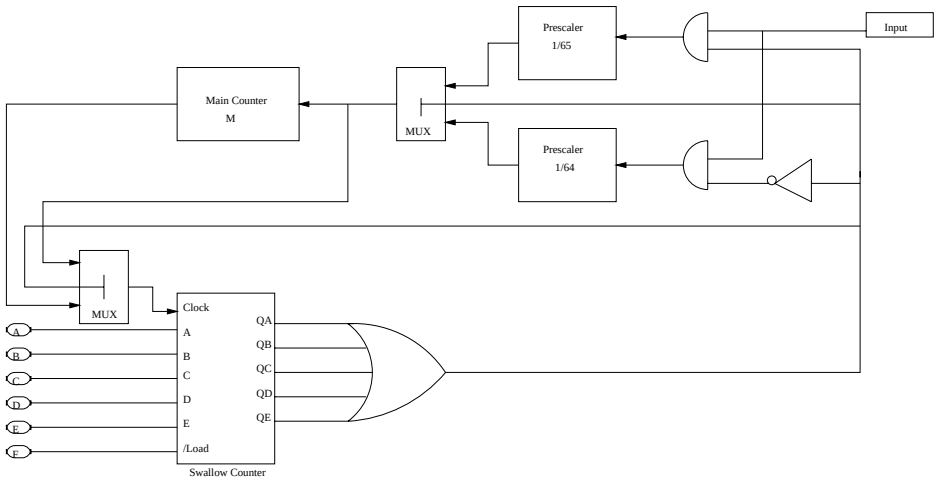

Figure 5.10: Control Logic

5.5 Output Interface

It is not discussed yet, that how the output of the PLL can be interfaced with a transceiver chip, in case the designed frequency synthesizer is supposed to be a part of it. Figure 5.12 shows, the interface of the pll with the rest of the circuit and some more details about a general transceiver's design architecture.

There are four inputs to the PLL synthesizer. Out of these four inputs, the three inputs are Data, LE, Clock, which are used to program the divider of the PLL, and the fourth one is the crystal oscillator's reference frequency. The output of the PLL is going into the 90 degree phase shifter. A phase shifter is a circuit which takes an input and generates two outputs, which are 90 degree out of phase with each other. In the configuration shown above, it is necessary that the output frequency is equal to the input frequency, but in most of phase shifter's designs , the output frequency

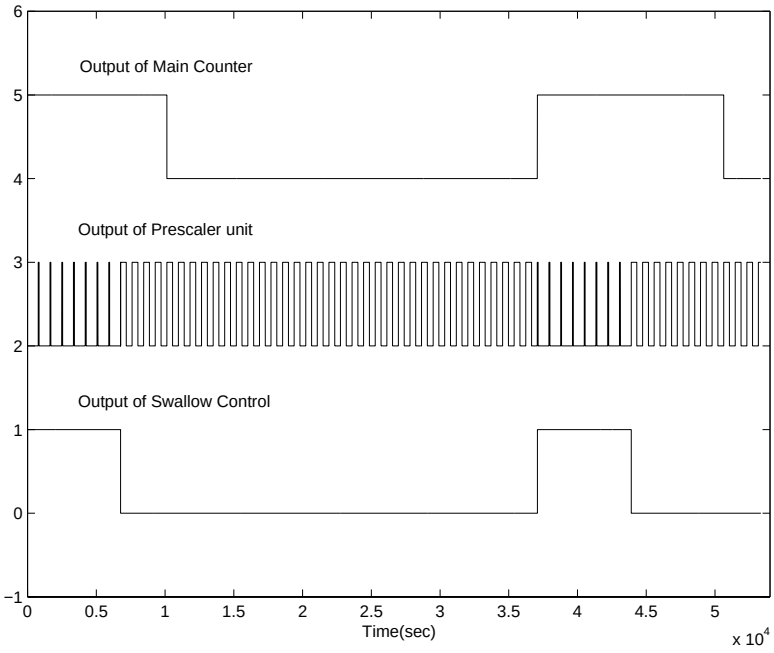

Figure 5.11: Simulink Results of Dual Modulus Programmable Divider Control Unit

reduces to half of the input frequency. To double the output frequency, a frequency doubler can be used . A phase shifters that can be used is shown in Figure refdquad

The Output1 and Output2 are 90 degree out of phase with each other and the output frequency can be doubled using a frequency doubler at the output of the PLL, so that the reduction of the frequency will be invisible for the rest of the transceiver circuit.

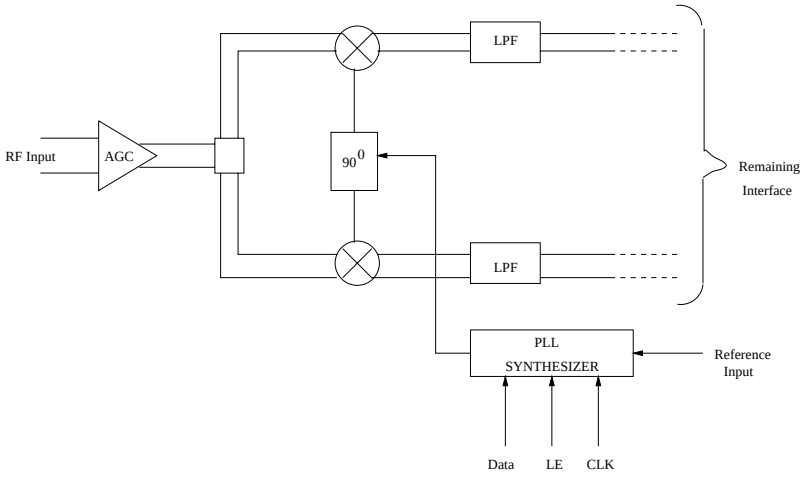

Figure 5.12: PLL Interface with a Tranceiver Chip

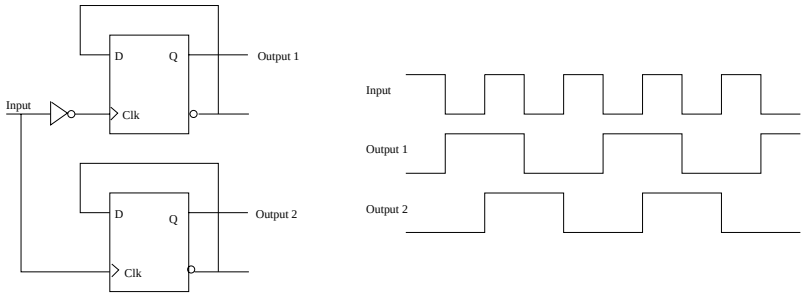

Figure 5.13: Quadrature Signal Generator and Output Waveforms

CHAPTER 6

CONCLUSION AND FUTURE WORK

Phase locked loop remained an interesting topic for the research, as it covered many discipline of electrical engineering such as Communication Theory, Control Theory, Signal Analysis, Design with transistors and opamps, Digital circuit design and non- linear analysis.

6.1 Summary

6.2 Future Work

The subject of phase locked loop is wide and diverse. There are many other aspects that can be combined in the design to achieve better performance and more powerful. Consider for instant incorporating Fault Tolerant Design Techniques to a PLL design. Since recent advances in VLSI technology has made it possible to put complex digital circuits on a single chip, more and more circuits are now combined on a single chip to make a system as compact as possible, such as a PLL in a transceiver chip. As a result of this capability, it is very hard to locate an error in the event if the output of a system is not the expected one. Some of such system are related to critical application, where it is necessary that the system operates reliably, even under the circumstances that one of the major component fails. The design techniques that

make it possible for a system to be operational even under the condition of failure are termed as Fault Tolerant Design Techniques and the system as the Fault Tolerant system.

So the idea of fault tolerance is that masking up the weak points of a system, where that system may become faulty under physical defects, environmental conditions or may be because of basic design errors. I would like to implement a Fault Tolerant Frequency Synthesizer, by including both digital design and VLSI fault tolerance techniques. The incorporation of this will make it more reliable and powerful.

The Other aspect that was not included in the MYDESIGN was the NOISE ANALYSIS. Since noise is an imporant parameter which affects the performance of a design mostly in non-linear fashion, is neccassry to accurately measure the performance of that design. Including noise consideration provide more details about the sensitive points and parametrs of a design. I would like to study the sources of noise in MYDESIGN and its effect on the performace as my future work.

APPENDIX A

SIMULINK MODELS

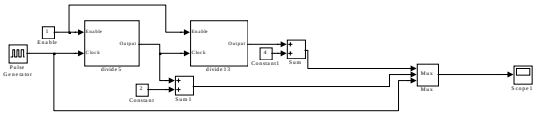

Figure A.1: Simulink Model of the Prescaler

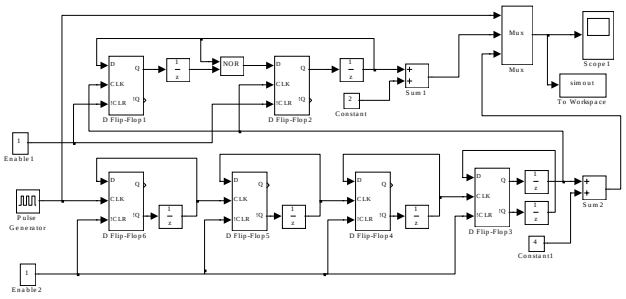

Figure A.2: Simulink Model of the Main counter

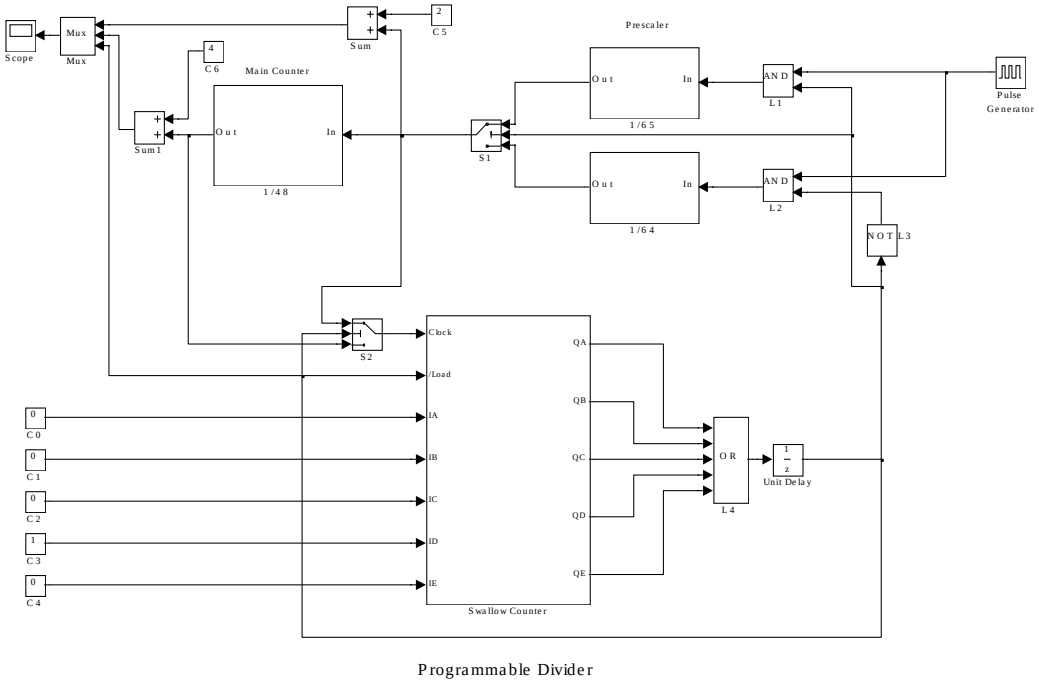

Figure A.3: Simulink model of Control Unit

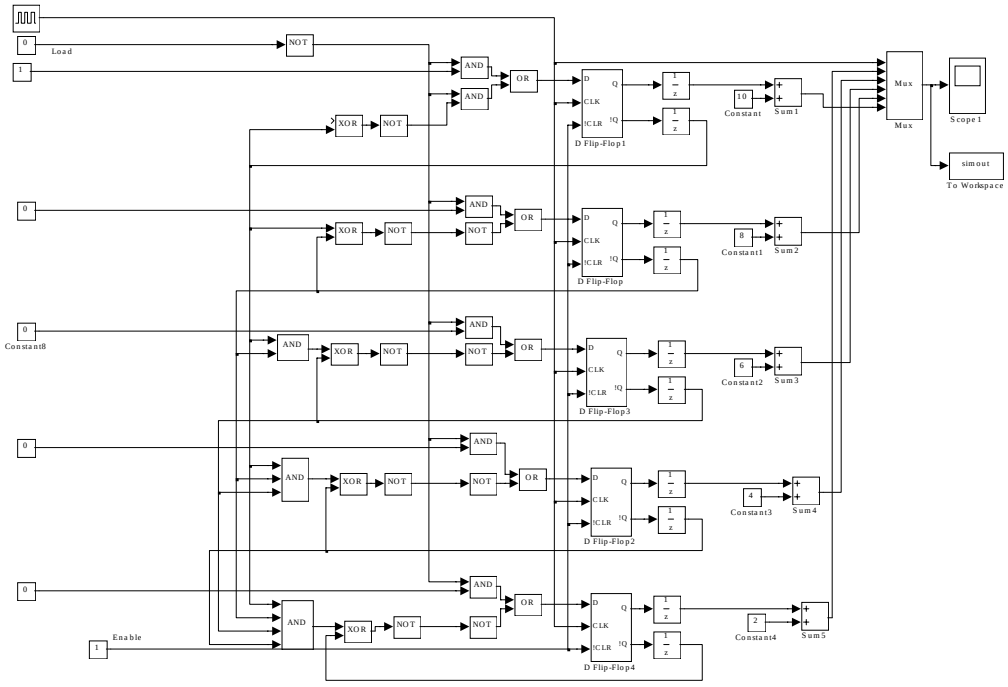

Figure A.4: Simulink Model of Down Counter

APPENDIX B

LIST OF SYMBOLS

K_o	VCO gain
K_d	Phase Detector gain
N	Frequency Divider Ratio
t	Real-time variable
$\theta_i(t)$	Phase of input reference
v_c	VCO control voltage
v_d	Phase Detector output voltage(average)
V_{dm}	Maximum value of v_d
V_{do}	Free running voltage of Phase Detector
v_i	PLL input voltage
v_o	PLL output voltage
θ_d	Phase difference between the reference and output signal
θ_e	Phase error between the reference and output signal
θ_i	Phase of PLL reference signal
θ_o	Phase of PLL output signal
ω_o	Output Frequency of VCO
ω_1	Frequency of loop filter pole less than K
ω_2	Loop filter zero frequency less than K
ω_3	Loop Filter pole frequency greater than K
$Z(h)$	High Frequency Gain of Loop Filter
$Z(s)$	Loop Filter Transfer Function

BIBLIOGRAPHY

[1] Roland Best. *Phase-Locked Loops: Design, Simulation and Applications.* McGraw Hill, Third edition, 1997.

[2] Robert H. Bishop. *Modern Control Systems.* Addison Wesley, Seventh edition, 1995.

[3] Gary Miller. *Modern Electronic Communication.* Prentice Hall, Fifth edition, 1996.

[4] James W. Nilsson. *Electric Circuits.* Addison Wesley, Fourth edition, 1993.

[5] Behzad Razavi. *RF Microelectronics.* Prentice Hall, First edition, 1998.

[6] John L. Stensby. *Phase-Locked LOOPS: Theory and Application.* CRC Press, First edition, 1997.

[7] Neil H. E. weste and Kamran Eshraghian. *Principles of CMOS VLSI Design : A System Perspective.* Addison Wesley, Second edition, 1992.

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