NUI MAYNOOTH

Ollscoil na hÉireann Má Nuad

ANALYSIS AND DESIGN OF HIGH ORDER DIGITAL PHASE LOCKED LOOPS

by

Brian Daniels, B. Eng., M. Eng.

A thesis presented to
THE NATIONAL UNIVERSITY OF IRELAND
in partial fulfilment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Electronic Engineering
National University of Ireland, Maynooth

December 2008

Supervisor of Research: Dr. Ronan Farrell
Head of Department: Dr. Frank Devitt

Table of Contents

List of Figures

List of Tables

ABSTRACT

The Phase Locked Loop (PLL) is an important component of many electronic devices; it can be employed as a frequency synthesizer, for clock data recovery, and as amplitude and frequency demodulators. It is an inherently nonlinear closed loop feedback system; the nonlinearity is due mainly to the fact that the feedback loop comparator exhibits quantization-like effects at its output. The consequence of this nonlinearity is a lack of understanding of the behaviour of the PLL loop, particularly the behaviour and stability of high order PLL systems.

This thesis presents a new design technique for high order Digital PLL (DPLL) systems with a charge pump phase frequency detector component, offering an alternative to the common design practice which is to analyze the DPLL using a linearised model of the analogue PLL. The linear model can only be justified for low order DPLL systems that are close to lock. This is due to the fact that as the DPLL systems loop order and complexity increases the linear model becomes increasingly inaccurate, thus high orders are considered risky. The benefit of a high order DPLL loop is a purer output signal with less jitter and therefore better spectral efficiency making them highly desirable.

The design of high order DPLL systems is realised here by utilising three novel design techniques. First, the complexity of the high order system equation is reduced by introducing an approximation of charge on the loop filter capacitors. Second, the nonlinearity is modelled using a piecewise linear model, thus the complexity of the model is further reduced by determining the system stability and lock time from only the first few samples in state space. Finally, to reduce the number of design variables that are required as the loop order is increased, filter prototypes, which only require one parameter, are introduced with the result of optimally placing the system poles. The consequence of implementing the above methodologies is that the mathematical restriction on the system order can be overcome and the stability of high order DPLL systems can be accurately determined.

ACKNOWLEDGEMENTS

First and foremost I would like to thank my wife, Suzanne, who has supported, encouraged and inspired me throughout this Ph.D., without her support it would not have been completed. I would also like to thank my supervisor, Ronan Farrell, who gave me the opportunity to do this Ph.D. and guided me through it during the good times and the bad. I would like to thank my two boys, Killian (3 years) and Oisín (1 year), who helped by keeping me on my toes, occasionally 'borrowing' my calculator and disordering my papers. A very special word of thanks for my parents, Martin and Mary, who supported me throughout my education and instilled in me the impetus to gain knowledge and who continue to support me throughout life. I would also like to mention a few other colleagues who have helped me in various ways; these include Ger Baldwin, Stephen Tigue, Paul McEvoy, Nigel Duignan, John Foody, and finally to my Family and Friends who have taken an interest in my work and cheered me on to completion – thank you.

LIST OF PUBLISHED CONTRIBUTIONS

- B. Daniels, R. Farrell, "Rigorous Stability Criterion for Digital Phase Locked Loops", ISAST Transactions on Electronics and Signal Processing Journal 2008, p1-10, No. 2 Vol. 3.

- B. Daniels, R. Farrell, "Nonlinear Analysis of the 2nd Order Digital Phase Locked loop", IET Irish Signals and Systems Conference, Galway 2008.

- B. Daniels, R. Farrell, "Design of Fourth Order Digital PLLs Using Filter Prototypes", IEEE Norchip Conference, Linkoping 2006.

- B. Daniels, R. Farrell, "Design Of High Frequency Digital Phase Locked Loops", Irish Signal and Systems Conference, Dublin 2006.

- B. Daniels, G. Baldwin, R. Farrell, "Modelling and design of high-order PLLs", SPIE Microtechnologies for the New Millennium Symposium: VLSI Circuits and Systems, Seville, 2005.

- B. Daniels, G. Baldwin, R. Farrell, "Arbitrary Order Event Driven Phase Lock Loop Model", IET – Irish Signal and Systems Conference, Belfast 2004.

CHAPTER 1

INTRODUCTION

1.1 Introduction

The Phase Locked Loop (PLL) is one of the most ubiquitous electronic components found in almost every electronic device from televisions to mobile phones, with a wide range of applications including frequency synthesis, clock data recovery, AM and FM demodulation, motor speed control, FSK decoders [1], and robotics to name but a few. The pervasiveness and popularity of the PLL is due to the robust nature and spectral purity of the PLL output signal, which are impossible to realise without the use of the PLL loop.

The performance of the PLL loop depends, in particular, on the loop filter characteristics. If the loop filter bandwidth is designed to be too narrow, for low noise performance, then the PLL loop will be slow to lock. Otherwise if the loop bandwidth is designed too wide for fast lock performance, then the filter may pass too much noise to the VCO causing the loop to become unstable. Ideally the designer needs to choose a loop filter bandwidth that provides the optimum lock time; the noise performance is achieved by increasing the filter order, and thus increasing the filter roll-off characteristic by an extra 20dB/decade per order. In general the most common system order is second, the reasons for this are: the relative simplicity of the design; the advanced field of linear PLL analysis that can be accurately applied to the design; and the fact that this linear analysis can be applied to all classes of second order PLL, including the Digital Phase Locked Loop (DPLL), the analogue PLL and the all-digital PLL. As the system order is increased the linear methods become significantly less accurate [2, 3]; this is due to the fact that each class of PLL system mentioned above is in fact nonlinear. The obvious solution to the design of such nonlinear PLL systems is to apply nonlinear methods to the PLL; however this solution is ineffective as it further complicates the analysis of an already complicated system, even for systems of low order. Thus linear analysis methods are almost exclusively used in practice.

The topic of this thesis is the analysis and design of high order DPLL systems, something that is currently not achievable. In order to enable the design of such high order DPLL systems two things need to be considered:

1. How can the nonlinearity of the DPLL be accurately modelled using a mathematically tractable analysis?
2. Is it mathematically feasible to extend this model to high order systems?

The high order DPLL analysis methodology that is presented in this thesis accurately models the DPLL nonlinearity by using a piecewise linear analysis. The complexities of the model equations are also reduced by introducing a novel charge approximation of the loop filter equations. The resulting nonlinear design methodology is mathematically tractable, informative and extendable to higher orders.

1.2 Structure of Thesis

In Chapter 2 of this thesis the PLL system performance and operation is outlined in detail, considering each class of PLL individually. Each component block within the PLL loop is presented, paying particular attention to the component blocks of the DPLL system. Finally the loop noise performance and the desirable properties of high order systems are presented. In Chapter 3 four different methods of modelling and analysing the DPLL are considered in detail, these are the linear approximation, nonlinear methods, circuit level simulation and finally event driven behavioural models.

This thesis proposes the following novel solutions to enable the design of high order DPLL systems:

1. The complexity of the DPLL system model equations is reduced by utilising a charge approximation of the loop filter equations. It approximates the differential terms in the system equations making it mathematically feasible to solve for high orders in closed form. This approximation is presented in Chapter 4.
2. To further simplify the loop analysis a piecewise linear state space mathematical analysis is proposed in Chapter 5. This technique determines the global stability boundaries for any order of DPLL.

3. In Chapter 6 both the charge approximation and the piecewise linear stability boundary analysis are used along with filter prototypes to produce a novel design methodology for high order DPLL systems.

In the final chapter of this thesis, Chapter 7, the conclusions that are drawn from the thesis are given, these conclusions will include avenues for future work that now exist due to the outcomes of the research summarised by this thesis.

CHAPTER 2

BACKGROUND

2.1 Introduction

The primary application of the Phase Locked Loop (PLL) is to generate a low noise signal with very precise frequency selection and stability. It achieves this by using a closed loop feedback circuit where the loop error is the phase difference between a low noise crystal oscillator reference signal and a locally generated signal, the feedback loop minimises this error, this has the effect of synchronising both signals. The benefit of such a scheme is that the resulting PLL output signal is robust (i.e. not prone to frequency drift) due to the high pass nature of the PLL loop and by including a frequency divider on the feedback path of the loop, the output signal becomes some multiple of the input and can be varied by varying the feedback divide ratio. Also, by utilising a PLL and in particular including a filter on the feed-forward section of the loop, the designer can choose the desired signal characteristics, such as low noise performance traded with a slow transient response. With such beneficial performance characteristics it is understandable that the PLL is such a common electronic component.

In this chapter an overview of the PLL system architecture and performance are presented, starting with a short review of the history of the PLL in Section 2.2. In the following section, Section 2.3, an introduction to the operation is given, considering each of the PLL components individually and then briefly discussing the common classes of PLL system in use, i.e. the digital PLL, the analogue PLL and the all-digital PLL. Arguably the most crucial component in the loop in terms of the system noise performance, transients and stability is the loop filter, this component is considered in detail in Section 2.4, paying particular attention to the loop filter parameters, the location of the system poles and zeros and how these may affect the loop performance. The loop filter structure is restricted by a number of system requirements; the traditional PLL filter structure is outlined in Section 2.4. In Section 2.5 the noise performance of the Digital PLL (DPLL) loop is considered, this

performance depends in part on the loop filter characteristics. In terms of the PLL system noise, particular attention is given to sources of the noise within the loop, the characteristic frequency of each noise source and the optimum means of attenuating this noise.

As mentioned, the loop filter is one of the more crucial PLL loop components in terms of the system performance. The choice of filter bandwidth, filter order and system gain will ultimately define the overall performance of the PLL system. Ideally the loop filter should be of a high order, optimally attenuating the out of band noise with a high frequency roll-off factor; this however is a difficult task due to the stability issues that exist with these higher order systems. In Section 2.6, consideration is given to the unique properties of such high order DPLL systems, why such systems are advantageous, why they are not currently popular and the issues that may arise when trying to design stable high order systems.

2.2 History of the Phase Locked Loop

The Phase Locked Loop (PLL) was first discussed in literature as far back as 1919 by Vincent [4] and Appleton [5], who experimented with the synchronization of oscillators. It wasn't until 1932 that it became a mainstream electronics device when it was used as part of a simpler alternative to the popular, but somewhat complicated superheterodyne receiver. The alternative device became known as the homodyne or synchronous receiver; see Figure 2.1(a).

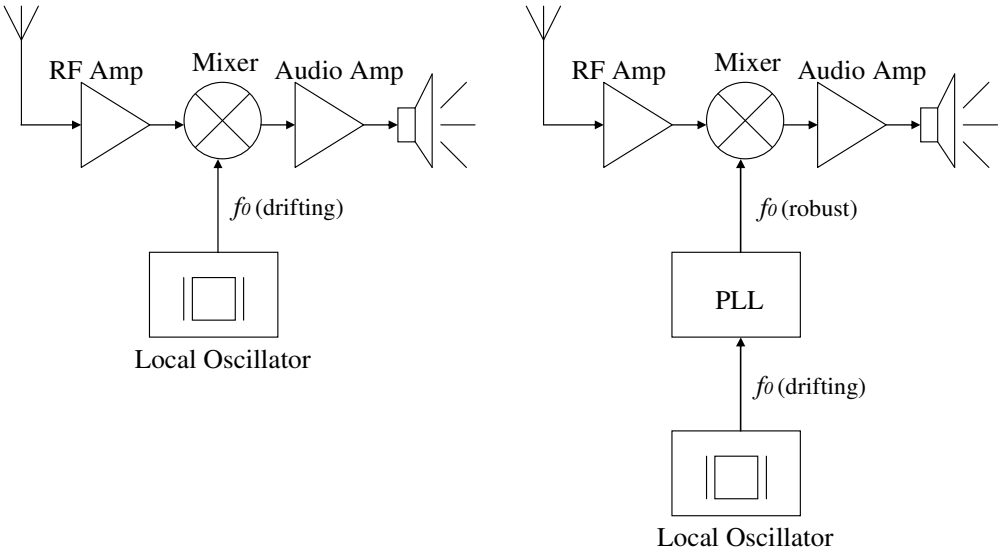

Figure 2.1 (a) Simple Homodyne Receiver (b) Homodyne Receiver Incorporating a PLL

The idea of the homodyne receiver was to tune a local oscillator signal to the desired input signal, and then demodulate the audio signal using a mixer and an audio amplifier. This device was simpler than the superheterodyne equivalent but had the drawback that the oscillator frequency, f_0, tended to drift in frequency and so the radio lost reception over a period of time. The solution to this frequency drift, as proposed by de Bellescise in [6], was to use a PLL to synthesize the oscillator signal, see Figure 2.1(b). This significantly reduced the frequency drift and thus improved the quality of the receiver output. However at the time that this simple alternative was conceived (1932) the cost of including a PLL in every receiver was significant and so the superheterodyne receiver continued to be used commercially until the PLL became available as a cheaper integrated circuit in the late 1960's.

While including a PLL in a radio receiver was considered expensive for the masses, there were still many uses for PLLs. One example, which achieved its first widespread commercial use was in the synchronization of the horizontal and vertical sweeps in television receivers [7]. Other earlier applications of the PLL was FM demodulation [8]. In the 1960's interest and publications in PLLs increased dramatically and culminated in the development of the PLL Integrated Circuit, which facilitated the rapid introduction of PLLs into consumer electronic devices [1].

Today, PLLs play an important role in modern communication devices. They are considered to be the most robust means of generating a low noise reference signal,

and the best means of determining the clock signal from a noisy source. There has been much research in this area and new applications and architectural advances continue to be found. The ubiquitousness of the PLL is due to its wide range of applications including frequency synthesis, phase modulation, clock data recovery, disk drive electronics, AM and FM demodulation, motor speed control, and FSK decoders [1], to name but a few.

The original PLLs developed by Vincent, Appleton, and de Bellescise [4-6] were purely analogue systems synchronising two analogue signals (the analogue PLL – APLL), later it was found that the PLL could also be use to generate digital signals by replacing some of the loop components, this class of PLL is known as the digital PLL. There are two types of digital PLL; these are the classical digital PLL (DPLL) and the all-digital PLL (ADPLL). The DPLL is the most common class of PLL due to its robustness, stability performance and high frequency characteristics. This thesis focuses on the classical DPLL. Each class of PLL is considered briefly in the next section.

2.3 Introduction to the Phase Locked Loop

The primary application of a PLL is to produce a low noise robust (no frequency drift) signal output. However considering that a robust signal can already be generated with the sole use of a crystal oscillator the question remains – why bother generating another signal by locking it to a noisy locally generated signal? To answer this question consider the demodulation of a radio signal (Figure 2.2).

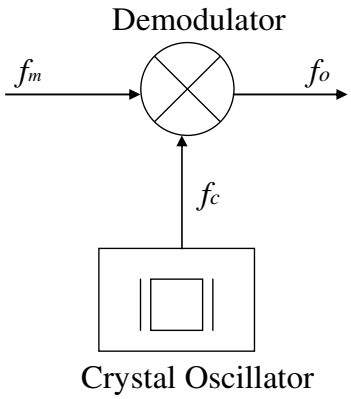

Figure 2.2 Demodulation using a Crystal Oscillator

A crystal oscillator uses the resonance of a vibrating crystal of piezoelectric material to create an oscillating signal with a precise frequency. The drawback of such a device is that the output frequency is fixed, so it is not possible to vary the carrier signal and tune in to another channel. The alternative to this is to use a voltage controlled oscillator (VCO) to generate the carrier signal (Figure 2.3).

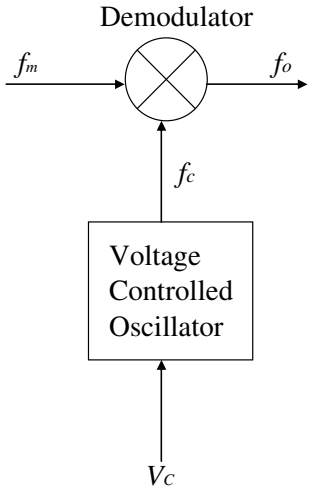

Figure 2.3 Demodulation using a VCO

The VCO generates an output signal with a frequency of f_c Hz; varying the VCO input voltage V_C results in a corresponding change in the VCO output frequency (and therefore the carrier frequency), this allows the demodulator to tune into other channels. The drawback of this setup is that the carrier signal f_c is vulnerable to noise on the V_C signal and any noise generated internally in the VCO device, thus this noise will be transferred discretely to the received radio signal. The channel may also tune out after a period of time as the carrier frequency drifts. The solution to both these issues is to use a combination of both the crystal oscillator and the VCO in the one feedback loop, as in Figure 2.4 below.

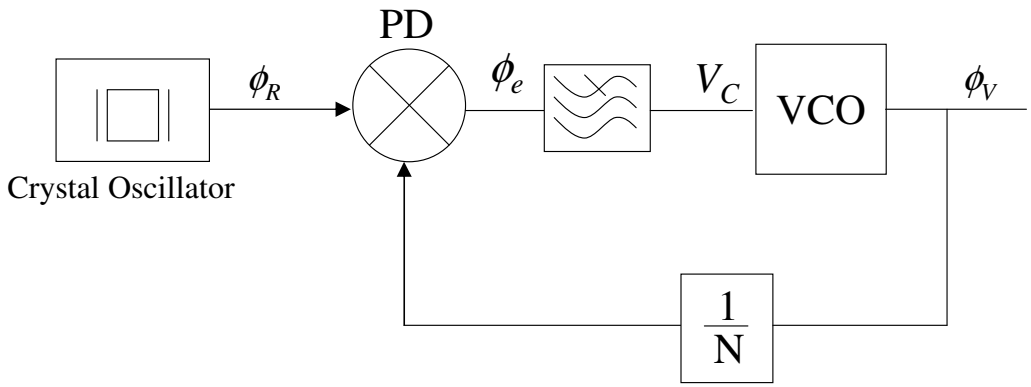

Figure 2.4 Phase Locked Loop Block Diagram

The feedback system structure shown in Figure 2.4 is known as a phase locked loop. It determines the phase and frequency difference between the input reference signal ϕ_R, generated by the crystal oscillator (with a frequency of F_R), and the VCO output signal ϕ_V (with a frequency of F_V). This difference, the loop error, ϕ_e, is reduced to zero by varying the VCO signal frequency, thus the loop error will be zero when the feedback and reference signals correspond in frequency and phase. If the VCO output signal drifts in frequency then an error will become apparent in the loop, this error has the effect of changing the V_C signal, driving the VCO output back to the desired frequency. In this manner the PLL keeps a steady output signal with no frequency drift. The VCO output frequency can also be varied by changing the feedback divide ratio N. When locked the output of the system will have a frequency of $F_R \times N$ hertz. By varying the divide ratio N the PLL output frequency can be varied while still maintaining the requirement of a robust and low noise output signal.

The PLL loop has stabilised, or is considered locked, when the error signal approaches zero or some stable equilibrium value. The performance of the loop outside of lock is not important, what is important is the manner in which lock is attained and maintained. This is measured using parameters such as lock time, capture range (also known as the acquisition or pull-in range), tracking range (or lock range), steady state error and stability. The capture range is the maximum value of the phase error within which an unlocked PLL will eventually reach the locked state, outside this range the loop will never attain lock. The tracking range, on the other

hand, is the maximum phase error offset for which a locked PLL loop will remain locked. These concepts are illustrated graphically in Figure 2.5.

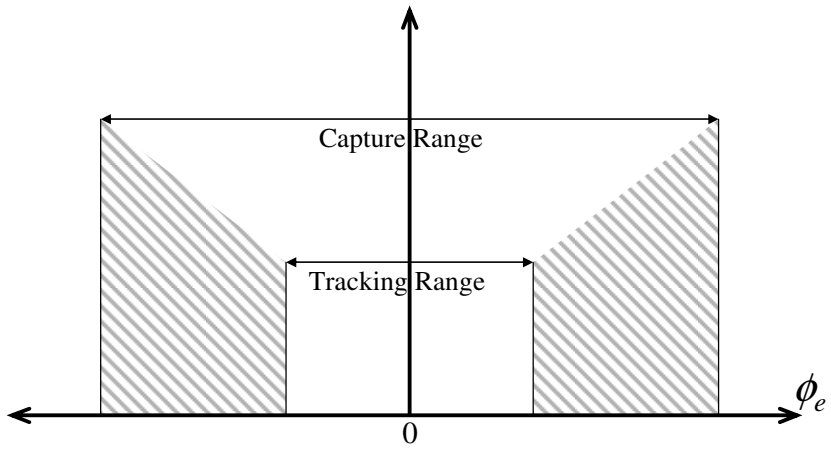

Figure 2.5 PLL Performance parameters

Due to control voltage noise, or frequency drift on the VCO output signal, both the VCO and reference signals will not always correspond exactly, but both signals will almost always be tracking each other. Therefore the phase error will rarely be exactly zero but will oscillate around zero within the tracking region of Figure 2.5. Outside the tracking region the PLL is unlocked.

2.3.1 Phase Locked Loop Components

There are a number of different classes of PLL systems, such as the analogue PLL (APLL), digital PLL (DPLL), and the all-digital PLL (ADPLL). All of these classes can be divided into a number of individual component blocks, these are the phase frequency detector (PFD), the loop filter, the voltage controlled oscillator (VCO), and a feedback divider (÷N). Each of these components may vary significantly depending on the particular class of PLL and on the particular PLL application. For example there are a large number of different types of PFD components, the designer selects one based on the:

- Type of input signal – analogue or digital.
- The output requirements of the PFD – i.e. will the loop error be represented by a voltage or a current value.
- The performance requirements of the overall PLL system – pull-in range, phase offset, cost, etc.

The choice of PFD in turn determines the requirements for the loop filter and the VCO; for example if a charge pump PFD is used, the output is a current source so it makes sense to utilize either a transconductance filter with a VCO, or a current controlled oscillator instead of a VCO.

2.3.1.1 Loop Filter Component

System noise is one of the most crucial design performance parameters of the PLL. Any system noise that exists on the control voltage node, shown as V_C in Figure 2.6 or that is generated by the VCO will cause unwanted frequency jitter on the output signal F_V. For this reason, the control voltage node is the critical node in the DPLL in terms of the system noise performance. The loop filter is a low pass passive filter and is an essential requirement for most applications to reduce the high frequency noise on the V_C node. The phase error signal at the output of the PFD may also contain noisy characteristics that are inherent to the PFD operation these are generally discrete-like high frequency components.

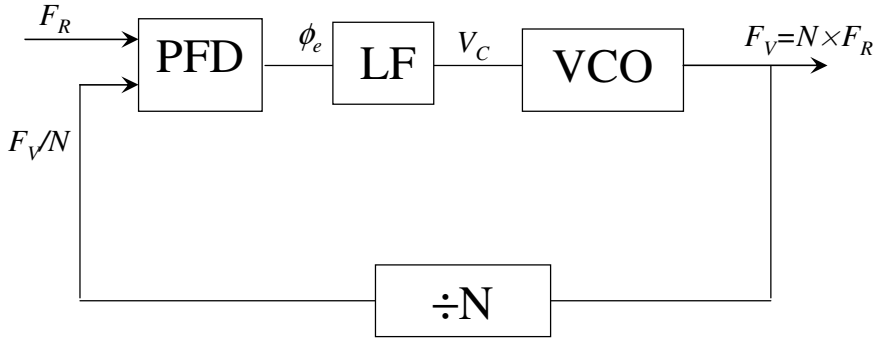

Figure 2.6 PLL System Block Diagram

The simplest form of PLL is a first order system with no loop filter, as described earlier and plotted in Figure 2.4; these systems are globally stable but noisy, producing large frequency jitter on the output of the loop which is intolerable for most applications. The solution to this unwanted jitter is to include a low pass loop filter (LF) before the VCO as in Figure 2.6.

The loop filter is generally of low order (first or second), its prime function is to isolate the sensitive control voltage node by attenuating the pre-filter noise. The loop filter noise performance can be enhanced by increasing the filter order; however this has the negative effect of increasing the loop complexity and potentially reducing the

stability. High order systems are rarely used in practice due to the fact that designers rely on linear methods to design the PLL which are not accurate enough to enable the design of high order PLL systems [2]; this is discussed in more detail in Section 2.6.

When designing the PLL loop filter, passive filters are traditionally chosen because active filters add excessive noise (from external power source of the op-amp) to the output of the filter [9, 10] and ultimately to the critical VCO control voltage node. This noise will be directly represented on the PLL output as frequency jitter, in some cases it is found to be excessive making the system unstable, and increasing the steady state error. However the exclusive use of passive filters adds some restriction on the choice of filter transfer function, thus it is possible to include both an active filter (for additional harmonic filtering) and a passive filter in the one loop. In the literature there are numerous papers that use only active filters in the loop; however it is very uncommon for these to be used in practice for the aforementioned reasons. The loop filter is considered in more detail in Section 2.4.

2.3.1.2 Phase Frequency Detector Component

The PFD block has the task of representing the phase and frequency difference between the two input signals F_R and F_V/N in voltage or current form. The exact operation of the PFD differs depending on the type of PFD; it may be a mixer (multiplier) phase detector, an EXOR gate, a charge pump phase frequency detector (CP-PFD), or an edge triggered JK flip-flop [1]. The mixer phase detector is used to determine the phase difference between two analogue signals, while the rest are generally used for digital signal comparison. The digital CP-PFD is more popular for a wide range of reasons including:

1. It produces a virtually infinite pull-in range [1].
2. It has a theoretical zero phase offset [11].
3. It can be implemented at multi-gigahertz frequencies [12].
4. It is low cost [13, 14] – this is at the expense of additional noise relative to the mixer phase detector [14]).
5. It can resolve phase differences in the $\pm 2\pi$ [10] (unlike the mixer or EXOR gate that can only resolve in the $\pm\pi$ range).

6. It allows the use of a passive loop filter while still being able to match the performance of a mixer phase detector using an active loop filter [15].

However no one PFD is best for all applications. For consistency this thesis considers exclusively DPLLs containing the most popular PFD component, the CP-PFD.

2.3.1.3 Voltage Controlled Oscillator Component

The VCO component block uses the loop filter output to generate an output signal with a frequency of F_V, this can be realised by a number of different methods, the most common of these is the LC tank oscillator which provides a much purer output signal than other VCO architectures [16]. The ideal VCO output frequency is determined using equation (2.1):

$$F_V = K_V V_C + F_{FR} \tag{2.1}$$

where K_V is the VCO gain, and F_{FR} is the VCO free running frequency (or the output frequency when the input control voltage V_C is zero).

2.3.1.4 Feedback Divider Component

As mentioned earlier the feedback frequency divider (÷N) is used in frequency synthesis to produce a PLL output signal frequency F_V that is some multiple, N, of the reference signal, F_R. There are two common classes of feedback divider: the integer-N and fractional-N dividers. The integer-N type divides the frequency by an integer divide ratio, while the fractional-N divider provides a fractional divide ratio. The fractional divide ratio is achieved by selecting between integer values, for example, if a divide ratio of 10.5 is required, then the fractional divider could divide by 10 for 50% of the time and by 11 for 50% of the time [17]. The variation in the divide ratio can be achieved by modulating the divider through the use of random jittering, phase interpolation, or a sigma delta modulator (SDM) [18]. The advantages of a finer frequency resolution with the fractional divide ratio are valuable in many modern applications. In terms of the analysis of the general PLL loop the sole effect of the divider (other than the introduction of some noise, which will be considered in Section 2.5), is to scale the feedback signal frequency F_V by a

factor of *N*. In this thesis the feedback divide ratio is chosen to be equal to 1 for clarity, however the divide ratio will be included in the presented equations.

2.3.2 Analogue Phase Locked Loop

The analogue PLL (APLL) generates a sinusoidal oscillating signal from an input reference analogue signal. All component blocks of the APLL are analogue as shown in Figure 2.7 with a PFD that is generally a mixer phase detector [19].

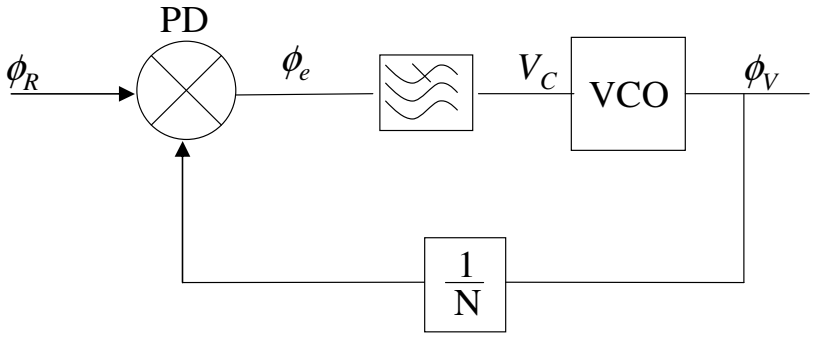

Figure 2.7 LPLL Block Diagram

A frequency or phase difference between the reference and VCO signals produces a phase error, ϕ_e, represented as a voltage at the output of the PD. The inherent nature of the feedback loop is to drive this phase error to zero. Thus the PD not only minimises the phase difference between the two signals, but also minimises the frequency difference.

The APLL is a non-linear device, due to the mixer PD, however it is generally linearised by approximating the PD to a linear subtractor, and approximating the VCO to some linear equivalent. This linear model is considered in more detail in Chapter 3 of this thesis.

2.3.3 Classical Digital Phase Locked Loop

The classical DPLL system differs from the APLL in that it generates a digital output signal as opposed to an analogue signal, i.e. the reference and VCO output signals are digital. The PFD compares these two digital signals using the CP-PFD component block, which has significant advantages over other possible PFD

implementations as discussed earlier. The block diagram of this DPLL system is shown in Figure 2.8.

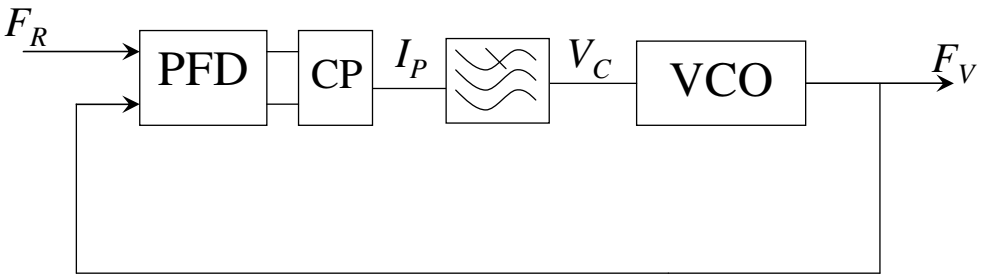

Figure 2.8 DPLL Block Diagram

The DPLL loop operation is similar to that of the APLL in that the information at the critical system nodes are the same, i.e. frequency at the PLL output and phase at the CP-PFD output. The CP-PFD determines the phase difference between the two input signals. If the phase and frequency of these input signals are equal then the output of the CP-PFD will be zero, otherwise the output will be some representation of the phase difference in either voltage form or, in the case of a CP-PFD, in current form. Thus it can be concluded that the CP-PFD operates in one of three possible states depending on the relationship between the two input signals, these are outlined in Table 2.1.

Input Signals	**CP-PFD State**
Reference leads VCO	Up
VCO leads Reference	Down
Signals Correspond	Null

Table 2.1 CP-PFD Input Signals and Corresponding System State

The CP-PFD changes state as falling or rising edge events[i] of the reference and VCO signals occur. The occurrence of these events determines whether the reference signal leads, lags or corresponds to the feedback signal. The CP-PFD state transitions and the events that cause these state transitions can best be described using the state transition diagram of Figure 2.9 below, where V↓ is a VCO falling edge event and R↓ is a reference signal falling edge event. Each state of the CP-PFD (Up, Down, or Null) yields a corresponding charge pump output current of $+I_P$, $-I_P$, or zero amps respectively, where I_P is the current gain of the charge pump and is equal to $2\pi K_P$.

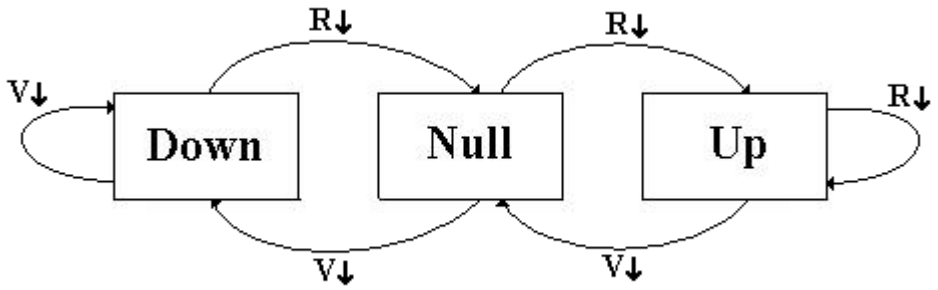

Figure 2.9 CP-PFD State Transition Diagram

To explain the operation of the DPLL consider an example, if the DPLL reference signal leads the VCO signal, then during any one period of the reference signal, *T*, an R↓ event will occur followed by a V↓ event. As long as this is the case the DPLL will oscillate between the Null and Up states and the CP-PFD will (on the average) output a positive current pulse during *T*. This positive current pulse has the effect of increasing the VCO control voltage, and therefore increasing the VCO output

[i] Note that in this thesis the falling edge convention is used exclusively.

frequency (pushing the V↓ event closer to the R↓ event), pushing the DPLL towards lock. Likewise if the VCO signal leads the reference signal, then the CP-PFD will oscillate between the Null and Down states resulting in a negative current pulse output, this decreases the VCO output frequency and again pushing the loop towards lock. The transition between states in the DPLL results in the nonlinear quantization or discrete-like effects that were described earlier. This state transition diagram will be used later as part of the behavioural modelling methods of Chapter 3 and ultimately as part of the proposed piecewise linear stability methodology of Chapter 5.

In addition to pre-filter noise attenuation, the DPLL with a CP-PFD also requires the loop filter to convert the discrete-like current output of the CP-PFD to a continuous voltage for operation by the VCO. This can be achieved by including a low pass first order RC filter before the VCO in the loop as in Figure 2.10.

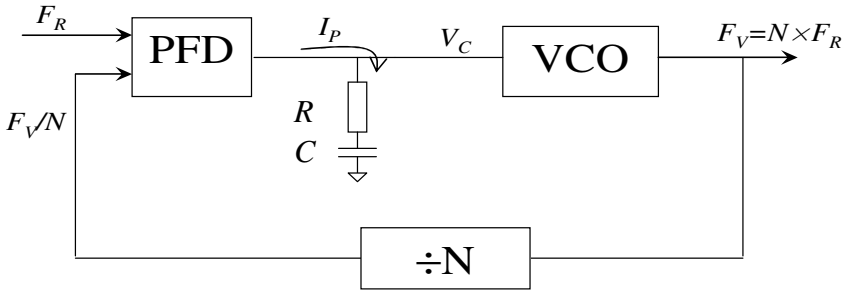

Figure 2.10 DPLL System Block Diagram

In the case of the DPLL, discrete V_C voltage jumps of value $I_P R_2$, where I_P is the current pumped into the filter and R_2 is the loop filter resistor, still exist due to voltage jumps across R_2, these are commonly attenuated by including an additional ripple capacitor in parallel with the loop filter, thus increasing the PLL order to third. As an example consider a PLL with loop component values of R_2 equal to 10 kΩ, C_2 equal to 100 nF, and an I_P equal to 0.1 μA, thus it is expected that the RC jumps will be equal to 1 mV. Figure 2.11 graphically illustrates these inherent RC jumps for this second order PLL system.

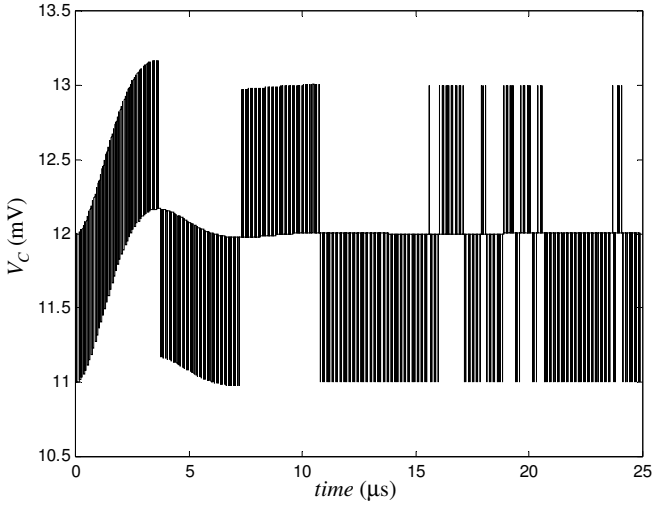

Figure 2.11 Second Order PLL Output with Inherent RC Spikes

By including an additional ripple capacitor in the loop filter it is expected that it will remove these RC jumps. Therefore by including an additional 100 pF capacitor this produces the less jittery output signal as plotted in Figure 2.12.

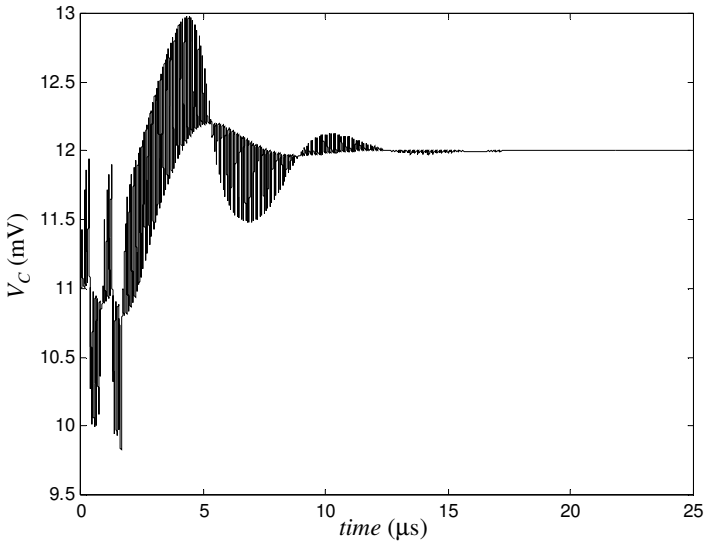

Figure 2.12 Third Order DPLL Output with Additional Ripple Capacitor

2.3.4 All Digital Phase Locked Loop

The classical DPLL of the previous section is a bit of a misnomer in that it has some analogue components. It is digital because it operates on a digital square wave signal. The all digital phase locked loop (ADPLL) differs from the classical DPLL

in that it has all digital components. The digital PFD output information is not an analogue voltage representation of the phase difference but is considered to be digital, containing either pulses or multi-bit values, as in Figure 2.13. In recent times some authors in the literature have referred to the ADPLL as the DPLL. This obviously leads to confusion; the traditional and the generally accepted definitions of the DPLL and ADPLL are as they have been described in this section.

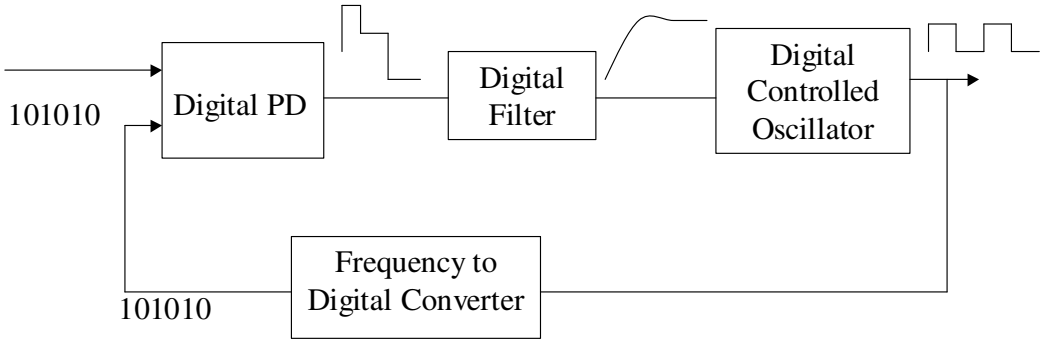

Figure 2.13 ADPLL block diagram

Though ADPLL systems are quite extensively researched and published [20-22] they are rarely used in practice, as serious issues still exist with their accuracy. Process variations in excess of 30% [23] can exist seriously degrading the phase noise of the system. Both the DPLL and ADPLL have similar applications; i.e. they both operate on digital signals, however the DPLL is considerably more robust, with less noise and can utilise more advanced analysis and design methodologies.

2.4 DPLL Loop Filter

The PFD and VCO components, as described in the previous sections, perform vital operations within the overall loop. The ideal performance of both these components can be determined by considering one parameter, i.e. K_P (the charge pump gain) and K_V (the VCO gain) respectively. These parameters are generally selected by rule of thumb; this is a relatively straight forward process. In contrast the ideal loop filter performance is determined from a number of design parameters. Thus selecting the loop filter parameters is more involved than the PFD and VCO rule-of-thumb methodology. This is because the loop filter characteristics will ultimately define the

stability and transient performance of the loop. More specifically the loop filter has two major responsibilities in the DPLL loop:

1. It is responsible for rejecting unwanted frequency components in the PFD output, which insures a low noise VCO control voltage.
2. It also converts the discrete-like output of the PFD to a continuous current signal.

When selecting the loop filter parameter values the PLL loop transient performance needs to be considered.

The number of loop filter parameters depends on the order and structure of the loop filter. The filter is commonly of first or second order. DPLL systems containing a first order filter exhibit two significant drawbacks: the RC spikes on the output that were discussed in Section 2.3; and the fact that the loop filter damping factor, the loop bandwidth, and the loop gain are all directly related. The loop bandwidth is a measure of the rate of convergence of the feedback loop, it is also known as the natural frequency $(\omega_n/2\pi)$ or the frequency at which the open loop gain equals unity. If the loop gain is increased (loop bandwidth increased), this causes an unwanted increase in the static phase error in the loop. If the loop gain is decreased (to reduce this static phase error) then the transient response of the system degrades, resulting in a slower lock time or, in the worst case, instability. Independence between these parameters can only be achieved by increasing the order of the filter, this means that the loop lock time and the loop noise or output signal jitter are related and thus a fast lock low noise DPLL cannot be achieved. As both these design parameters depend on the choice of the loop filter cut-off frequency, ω_c, and are contradictory, the choice of ω_c becomes a difficult task for any designer. The only guideline on the choice of ω_c is that it should be chosen to be no greater than one tenth of the reference signal, based on this the DPLL is designed as follows:

1. The loop filter parameters are selected based on the desired system bandwidth, ω_c.
2. The VCO and PFD gain components are selected using 'rule of thumb'.
3. The PLL system is simulated to check the system transients and noise band limit.
4. The loop bandwidth, ω_c and the gains are reselected to vary the system transients and band limit as required, until the optimum system is found.

5. The system is simulated again (return to 3) until the desired system performance is achieved.

It is obvious from this design methodology that the initial choice of ω_c is crucial to the overall design process, if a bad selection of ω_c is made initially then an excessive amount of circuit level simulations may be required.

The DPLL loop filter transfer function generally has one zero and *W-1* poles, where *W* is the DPLL system order. An additional non-filter pole exists due to the integrator in the VCO; the single system zero in the filter is due to the transconductance nature of the filter. All of the additional poles are due to the denominator of the loop filter transfer function. In the case of the second order DPLL, it places the system poles in the locations as shown in Figure 2.14.

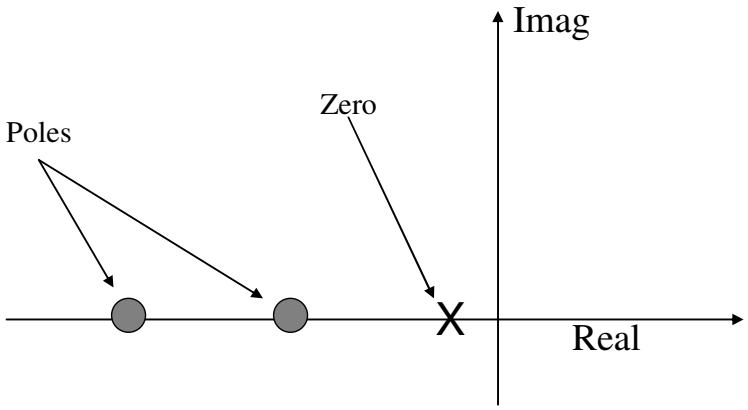

Figure 2.14 Plot of Second Order Poles

No consideration is given to the pole locations at all, despite the fact that it can give a good indication of the response of the linear system. The expected impulse response for various pole locations is shown in Figure 2.15.

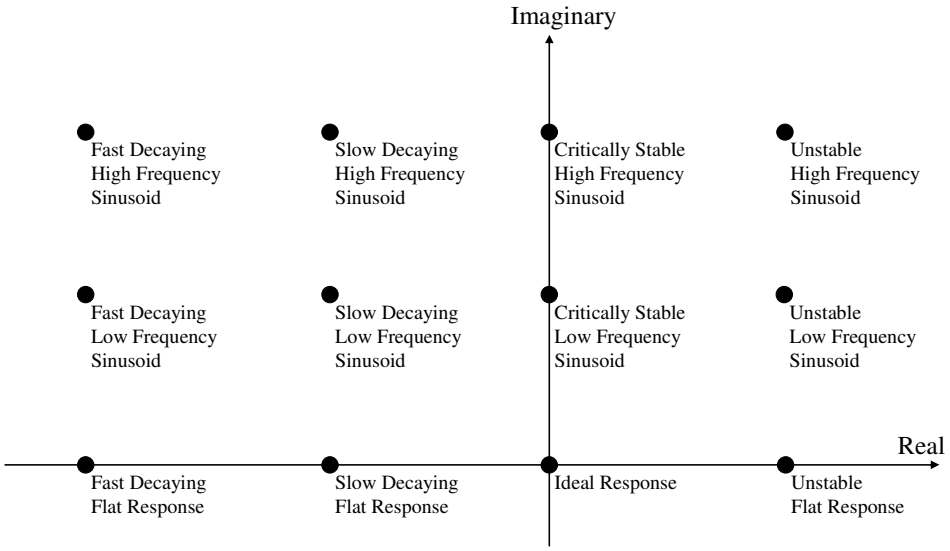

Figure 2.15 Effects of Loop Filter Pole Location

The pole location determines the systems transient response, while the zeros establish the relative weightings of the system poles. For example, moving a zero closer to a specific pole will reduce the relative contribution of that pole. The second order DPLL zero is generally placed close to the origin relative to the system poles. All higher order systems have only one zero placed in a similar location to the second order system, as the order is increased the system has additional poles, some of which become complex conjugate. These pole locations are outlined in Figure 2.16.

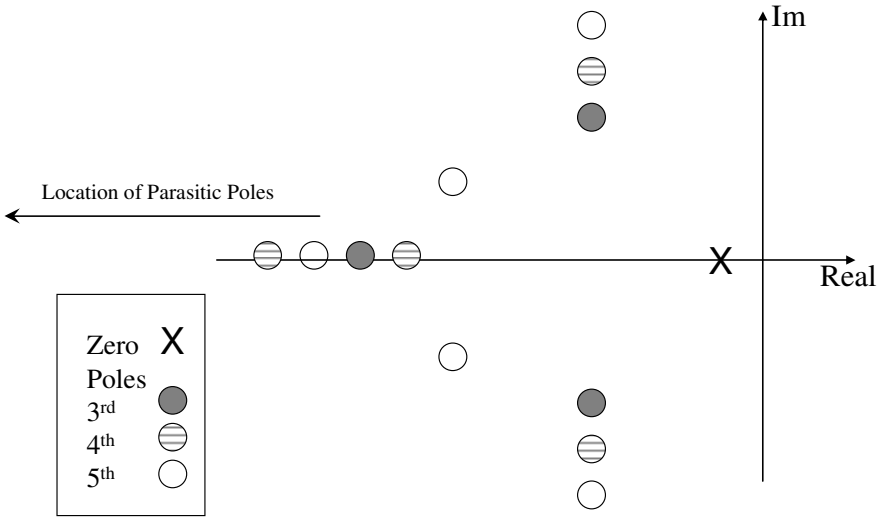

Figure 2.16 High Order Pole Locations

The traditional second and third order DPLL loop filter may include additional parasitic poles[ii] to achieve a sharper cut-off characteristic. Such high order systems are difficult to stabilise due mainly to the lack of an accurate analytical design methodology for such systems, and therefore the difficulty in determining a set of system parameters that will result in a stable system. Most PLL systems are third order topologies with the third pole being much further from the origin than the other two; this provides a relatively stable system with adequate noise suppression. A solution to the problem of increasing attenuation in the stop band can be provided by high order loop filters.

Designing the DPLL system using pole placement, can provide good system performance, however traditional design methodologies stay clear of choosing individual pole locations and use the filter cut-off frequency design parameter ω_c and some knowledge of the system stability boundary to design the PLL. If ω_c is chosen too large relative to ω_R (i.e. the loop bandwidth is selected to be too wide), then this may result in degradation in the loop performance. Alternatively, choosing a small ω_c relative to ω_R, will have a slower acquisition time. Obviously some trade-off is required so that the system produces low noise and fast acquisition time through the optimum choice of ω_c. The traditional choice of ω_c from existing examples in the literature [13, 24-26], are plotted in Figure 2.17 against the normalised system lock time ($t_{LCK}F_R$).

[ii] The parasitic poles in this case are poles that are placed significantly away from the system cut-off frequency ω_c as in Figure 2.16.

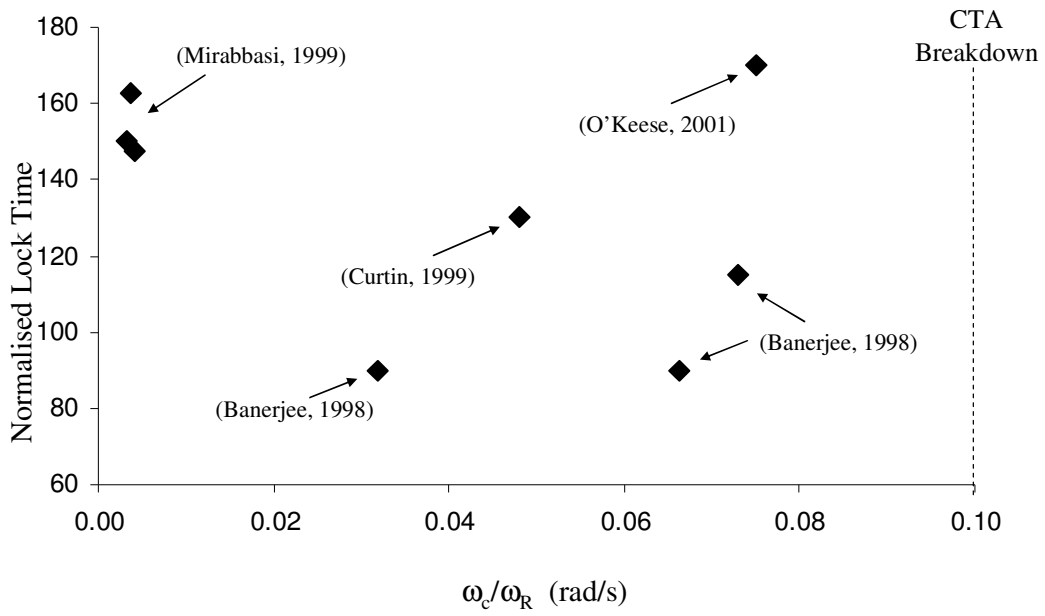

Figure 2.17 Plot of Traditional DPLL ω_C against Lock Time

If the DPLL is to have both a wide bandwidth and a fast acquisition time, then it is necessary to choose an ω_c/ω_R to the right of Figure 2.17, but to the left of the continuous time approximation[iii] (CTA) breakdown point [27]. For low loop noise, ω_c is required to be small and therefore to the left in Figure 2.17. Traditionally ω_c is arbitrarily chosen to be far away from the breakdown point; this tends to be significantly to the left as shown in Figure 2.17. It is clear that by identifying and designing a DPLL with a ω_c at the optimum point, will produce a loop response with maximum phase noise attenuation for the minimum requirement of lock time. The optimum choice of ω_c will be considered later in Chapter 6.

[iii] The continuous time approximation is also known as the linear approximation which will be considered in detail in Section 3.2. The breakdown point is the point where this approximation becomes invalid – when the filter cut-off frequency is greater than one tenth of the reference frequency. This is discussed in more detail in Section 3.2.1.

2.4.1 Loop Filter Structure

In Figure 2.6, shown earlier, the loop filter is represented by a modular block, this can be replaced by any type or order of passive loop filter. The traditional component structure of a first order transconductance loop filter is as given in Figure 2.18 below.

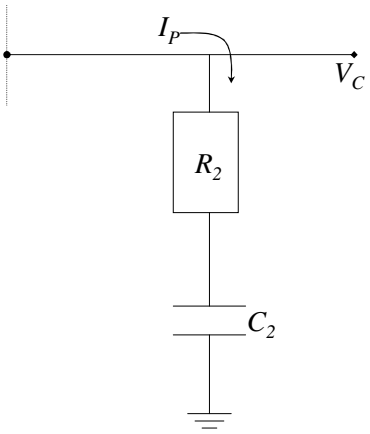

Figure 2.18 First Order Filter Structure

As discussed earlier the first order filter is the most common DPLL filter order due to its fast responsiveness, but this is at the expense of excessive frequency jitter on the DPLL output due to the inherent voltage jumps across resistor R_2. The solution to this jitter is to include an additional ripple suppressing capacitor in parallel with the RC branch, as shown in Figure 2.19, thus increasing the loop order to three.

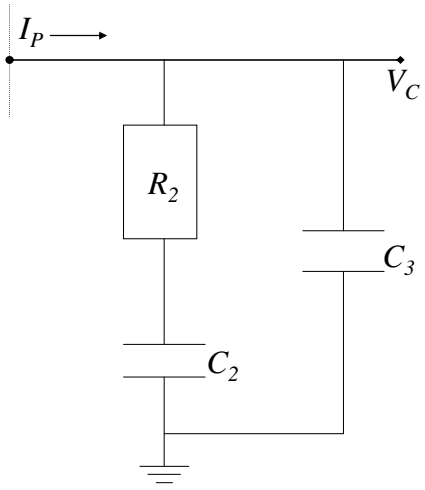

Figure 2.19 Third Order DPLL Loop Filter Structure

The C_2 capacitor of the second order filter, determines the settling time while the additional capacitor C_3 suppresses the RC spikes discussed earlier that is generated by the current pumped into the RC section of the filter. Additional spikes may arise due to mismatches between the width of the Up and Down pulses of the PFD, as well as charge injection and clock feed through. In order to maintain stability of this third order system the pole added by C_3 must remain below C_2 by a factor of ten to avoid under-damped settling.

Increasing the loop order to four is achieved by adding the additional components R_4 and C_4 as shown in Figure 2.20, the additional pole further suppresses the ripple on the VCO control voltage however the stability of the loop becomes more of an issue. Normally the additional pole due to C_4 is placed out of band so that it has minimal effect on the loop performance. One rule-of-thumb is that C_4 should be significantly less than $1/10^{th}$ of C_3.

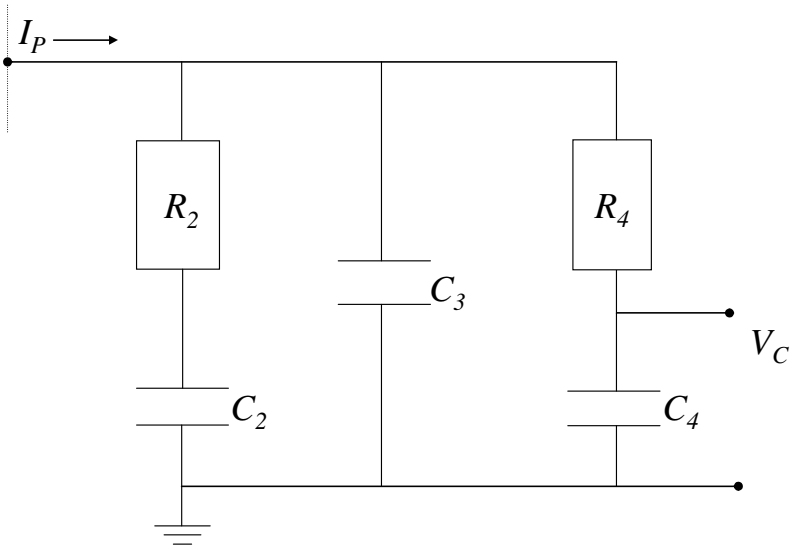

Figure 2.20 Fourth Order Loop Filter Structure

The primary advantage of such a higher order filter is that it will have a steeper frequency roll-off than a lower order filter (20dB/decade/pole [9]), and hence further attenuate the out-of-band noise.

As the order of the loop is increased the calculation of the loop filter transfer function becomes cumbersome. However this can be simplified by considering the generic nature of high order loop filters – i.e. all loop filters of order greater than four will

have a structure as shown in Figure 2.20 but with additional RC sections cascaded to the output of the lower order structure. Thus, using a two-port methodology the high order loop filter can be reduced to generic blocks. To achieve this consider the filter structure shown in Figure 2.21, where V_i and V_o are the input and output voltage respectively, and Z_1 and Z_2 are the component impedances for a second order filter.

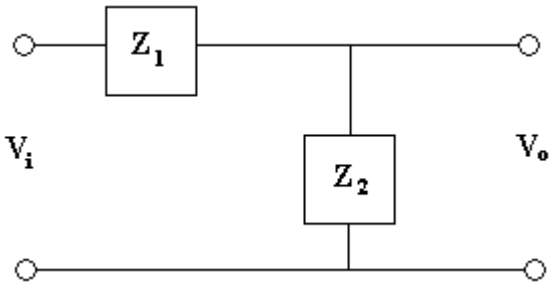

Figure 2.21 Simple Low Pass Filter

The transfer function of this generic filter system of Figure 2.21 is calculated as:

$$\frac{V_o}{V_i} = \frac{Z_2}{Z_1 + Z_2} \tag{2.2}$$

If Z_1 is a resistor and Z_2 is a capacitor, then a low pass filter is formed with the filter gain given by:

$$\frac{V_o}{V_i} = \frac{\frac{1}{sC_2}}{\frac{1}{sC_2} + R_2} = \frac{1}{1 + sC_2R_2} \tag{2.3}$$

The magnitude response is given by:

$$\left|\frac{V_o}{V_i}\right| = \left|\frac{1}{1 + sC_2R_2}\right| = \left|\frac{1}{1 + j\omega_c C_2R_2}\right| = \frac{1}{\sqrt{1^2 + (\omega_c C_2R_2)^2}} \tag{2.4}$$

The cut-off frequency where the magnitude of the gain is 0.707 is given by:

$$\omega_c = \frac{1}{R_2C_2} \tag{2.5}$$

Determining the transfer function of high order DPLL can be simplified by using this two-port design methodology and the generic transfer function of Z_1 and Z_2, this is outlined in more detail in appendix A. The transfer functions for the loop filter structures of Figures (2.18 – 2.20) are given in Table 2.2 below.

DPLL Order	Loop Filter Transfer Function
2nd	$\frac{C_2R_2s+1}{C_2s}$
3rd	$\frac{C_2R_2s+1}{C_2C_3R_2s^2+(C_2+C_3)s}$
4th	$\frac{C_2R_2s+1}{\left(\begin{array}{c}C_2C_3C_4R_2R_4s^3+(C_2C_3R_2+C_2C_4R_4+C_3C_4R_4+C_2C_4R_2)s^2+\\+(C_2+C_3+C_4)s\end{array}\right)}$

Table 2.2 Filter Transfer Function for the DPLL.

2.4.2 Loop Filter and System Bandwidth

As discussed earlier, the loop filter bandwidth, ω_c is inversely proportional to the system response time and directly proportional to the noise attenuation characteristics, thus to reduce the jitter on the PLL output a narrow filter bandwidth is required which is consistent with the system response requirements. This noise/response-time relationship can be illustrated by taking two second order linear PLL system examples, $H_1(s)$ and $H_2(s)$, where the closed loop transfer functions are chosen as in equations (2.6) and (2.7), respectively.

$$H_1(s)=\frac{12x10^{-4}s+100}{7.54x10^{-9}s^2+12x10^{-4}s+100} \tag{2.6}$$

$$H_2(s)=\frac{9x10^{-4}+100}{3.77x10^{-9}s^2+9x10^{-4}s+100} \tag{2.7}$$

The coefficients of the above system transfer functions are chosen such that they are low pass in nature and so that they have different bandwidths, as shown in Figure 2.22. The exact calculation of the bandwidths of systems $H_1(s)$ and $H_2(s)$ from Figure 2.22 can be made by finding the frequency at which the magnitude has

decayed by 3dB from the maximum. These are found to be approximately $1x10^5$ rad/sec, and $2x10^5$ rad/sec respectively.

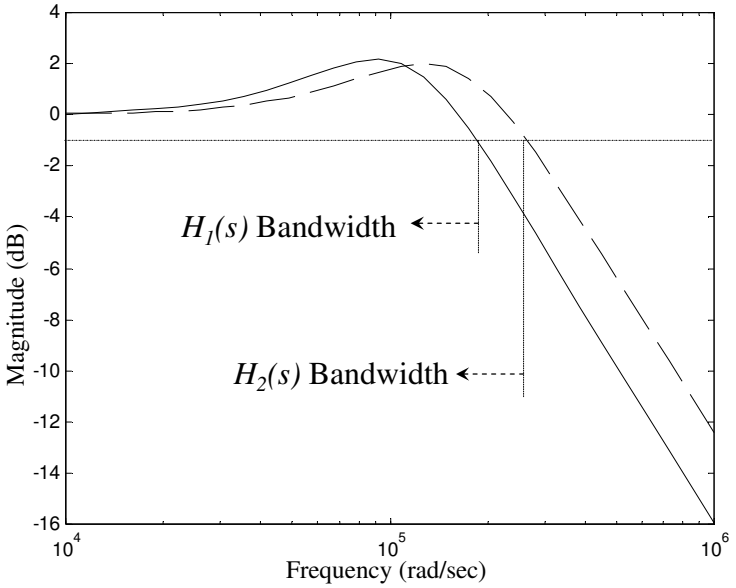

Figure 2.22 Magnitude Plots of $H_1(s)$ and $H_2(s)$

From the theory we would expect that wider bandwidth system $H_2(s)$, should have the faster response time at the expense of degraded noise performance. This can be illustrated by plotting the transient response of each system to a step in the phase error, as shown in Figure 2.23.

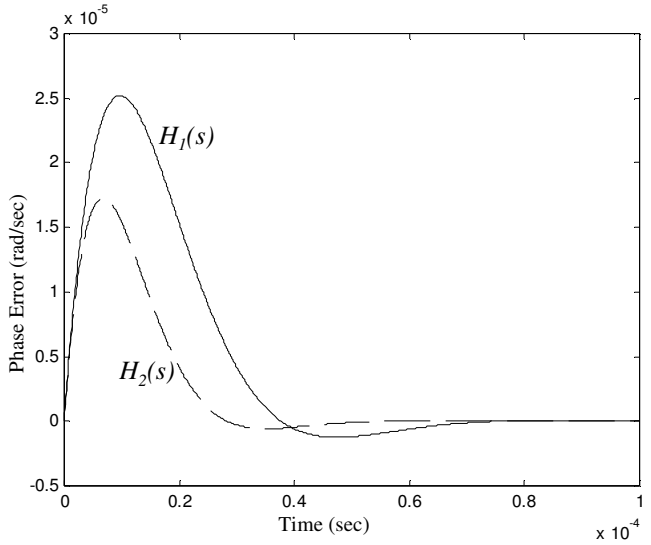

Figure 2.23 Time Domain Response of Systems $H_1(s)$ and $H_2(s)$

As expected the wider bandwidth system, $H_2(s)$, provides a faster response, however due to the narrower bandwidth of $H_1(s)$ it exhibits better roll-off characteristics and will pass significantly less noise to the VCO relative to $H_2(s)$, this illustrates the significance of the choice of ω_c in terms of the best design of the DPLL. The noise band limit is considered in more detail in the next subsection.

It is worth noting at this point that a number of alternative DPLL stabilisation methods exist that achieve both fast lock and low noise by varying the loop filter bandwidth ω_c. These work by using advanced architecture techniques such as gear-shifting [28, 29], and aided acquisition dual-loops [30, 31]. Gear-shifting varies ω_c from a fast locking wide bandwidth loop when the PLL is out of lock, to a low noise narrow bandwidth loop when loop lock is detected. Similarly dual-loops achieve fast lock and low noise using two feedback loops with contradicting bandwidths. Another solution suggested by Land [32] uses a sample and hold circuit that operates at double the input frequency. This reduces the unwanted ripple component without decreasing the bandwidth of the loop filter.

2.5 DPLL System Noise Characteristics

One of the most crucial design parameters of the PLL is the loop noise performance. To explain how the loop noise on a PLL output can effect the performance of a system consider an example of a generic RF front-end of a GSM mobile phone, the block diagram is given in Figure 2.24.

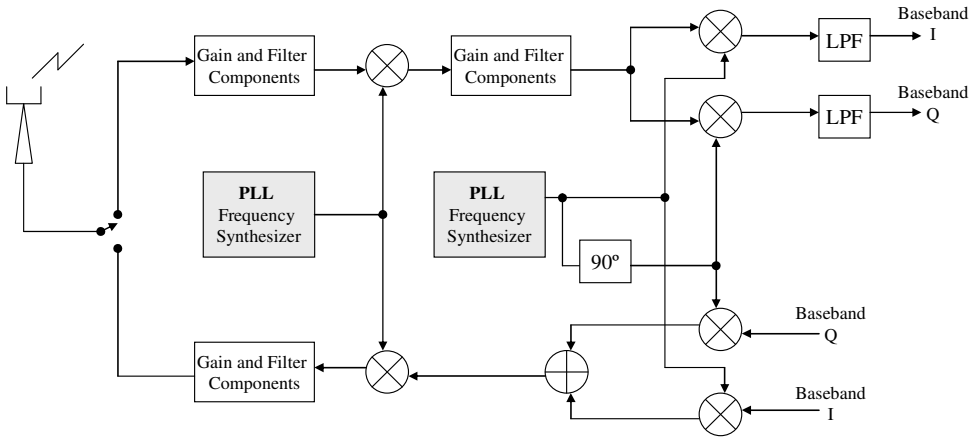

Figure 2.24 Generic RF Front-End

On the receiver side of the RF transceiver the received signal is amplified, filtered and then passed to the mixer section for demodulation of both the in-phase (I) and quadrature (Q) components, in effect the PLL is generating the local oscillator (or carrier) signal for the demodulation process. Likewise the PLL is used on the transmitter side as the local oscillator for the modulation process. Consider the case where there are two signals (one of these signals is unwanted) with frequencies ω_1 and ω_2 radians/second, both being received at the antenna, as in Figure 2.25. These signals are again amplified, filtered and demodulated down to baseband with a noisy local oscillator signal generated by the PLL. The resulting demodulated signal is shown in the bottom plot of Figure 2.25.

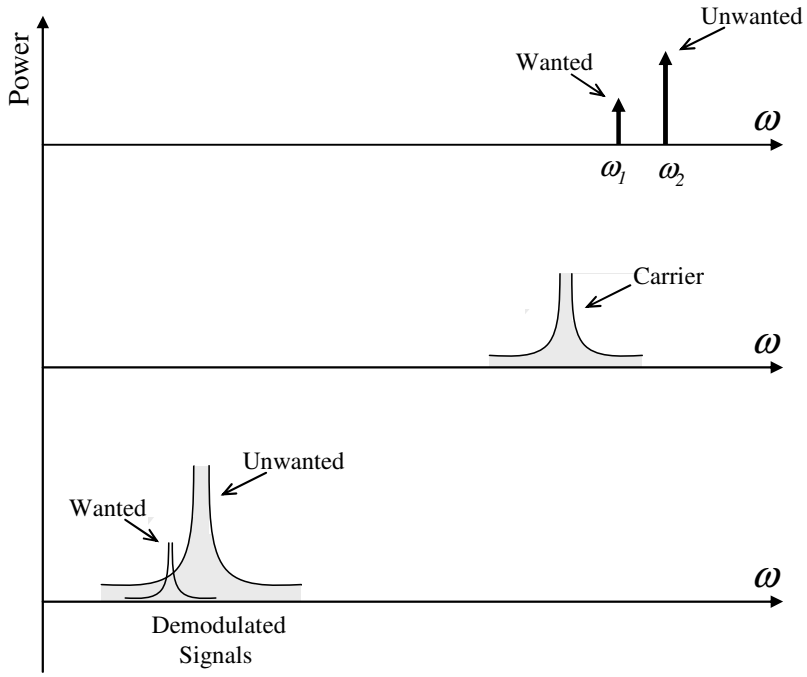

Figure 2.25 Reciprocal Mixing of Adjacent Channels

The desired signal ω_1 is corrupted by the unwanted signal in an adjacent channel. These local oscillator signal skirts affect the channel selectivity of the transceiver, this is known as reciprocal mixing[33]. The noise performance of the PLL as the local oscillator is crucial, the better the noise performance the closer the adjacent channels can be spaced without reciprocal mixing. This has the benefit of better spectrum efficiency. In terms of the noise skirts of Figure 2.25 the VCO is the biggest contributor relative to other noise sources, these undesirable noise skirts can

be reduced in a number of ways: using a VCO with a lower gain; designing a higher order loop; or adding a band-stop filter to attenuate these tones.

For a predefined VCO component the overall PLL noise performance is essentially set by the loop filter characteristics; this filter is designed mainly with the purpose of attenuating the out-of-band noise within the loop. To achieve this it is desirable to have a loop filter that is low pass in nature, as can be seen in the frequency plot of Figure 2.26 below.

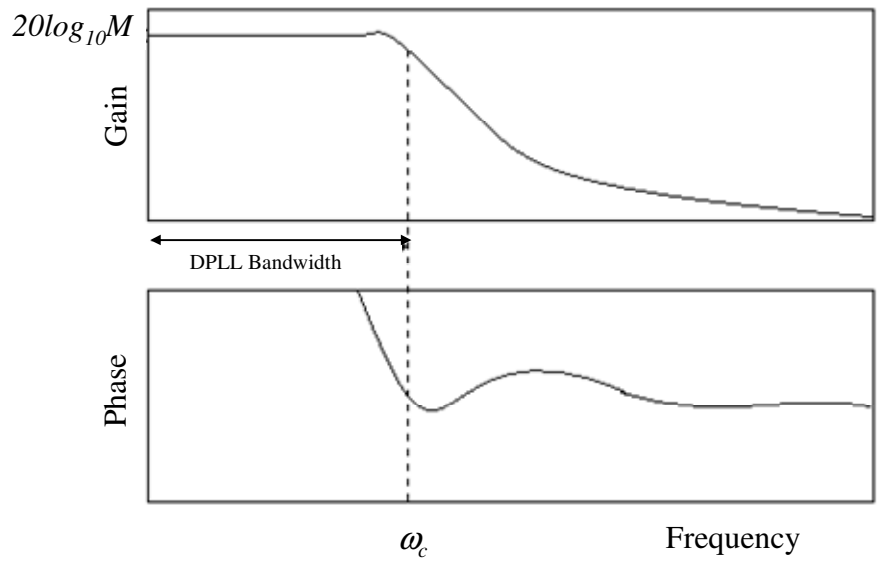

Figure 2.26 Frequency plot of the Loop Filter

Ideally the loop filter attenuates all unwanted disturbances from the control voltage signal that may interfere with the loop carrier signal and ultimately give a purer output signal. However this is generally not always the case as some noise will inevitably pass through to the VCO and generate some form of jitter on the loop output, it is hoped that with good design practice the control voltage, V_C, noise can be made insignificant. The linear model is traditionally used to model this noise, however the linearising assumptions that are made (the DPLL linearization is considered in detail in Chapter 3 of this thesis) assume that the high frequency product terms will be attenuated by the loop filter and are therefore ignored in the analysis, by including these high frequency product terms the spectral purity of the output signal is significantly improved [34], however the system equations are no longer linear.

In conclusion, other than using high quality and low noise loop components the best way to minimise the loop noise is to design the filter with an optimum cut-off frequency, to achieve this we need to determine some knowledge of the noise sources within the loop. There are four major noise sources in DPLL circuits; these are the crystal oscillator reference phase noise; the PFD noise; the VCO noise and the feedback divider noise. Each of these sources of noise are considered in the next subsection.

2.5.1 Sources of Noise within the PLL Loop

A block diagram of the DPLL noise sources is shown in Figure 2.27 below [35]. In this section each of these noise sources is considered in detail.

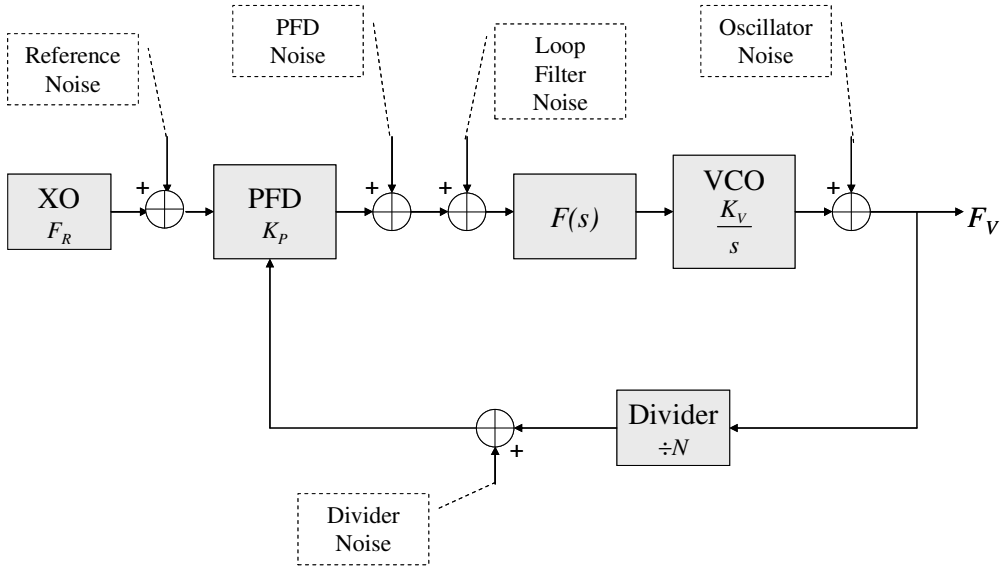

Figure 2.27 DPLL Noise Sources

As mentioned earlier the DPLL loop filter ($F(s)$) is low pass, therefore any noise in the DPLL loop with a frequency less than ω_c is multiplied by a factor of K, where K is the system gain. Noise with a frequency of greater than ω_c is attenuated by an additional rate of 20dB/decade/pole, in order to minimise this noise the loop bandwidth must be as small as possible. Banerjee [24] suggests that the ideal reference frequency should be at least 50 times that of the filter cut-off frequency ω_c. However this excessively narrow bandwidth has the effect of increasing the lock time, limits the capture range, and degrades the stability of the loop. In contrast the VCO has a high pass filter characteristic, it tends to contribute more noise within the

loop bandwidth where the loop bandwidth is narrow (less than the theoretical optimum loop bandwidth), these tones are attenuated in the loop filter. In order to minimise the VCO phase noise contribution the loop bandwidth must be maximised, this creates a conflict when trying to minimise both sources of noise.

The crystal reference phase noise and PFD noise are amplified within the loop bandwidth ω_c, and attenuated outside this, at offsets much greater than ω_c the dominant noise source is the VCO oscillator phase noise. Close to ω_c the noise contributions are a combination of all these noise sources. Furthermore spurious tones occur in the DPLL output due to component leakage, divider inclusion, dead zone elimination circuitry, mismatch in the charge pump PFD and many other factors[15, 36]. PFD mismatch occurs when the charge pump sink pulse $t_{sink}I_{sink}$ is not equal to the charge pump source $t_{source}I_{source}$, shown in Figure 2.28. This occurs because the charge pump is not perfectly balanced; the NMOS transistor that sinks current may have half the turn on time of the PMOS transistor that sources current, in this case the CP-PFD tends to be the dominant noise source in the DPLL. This noise level is directly proportional to the reference frequency.

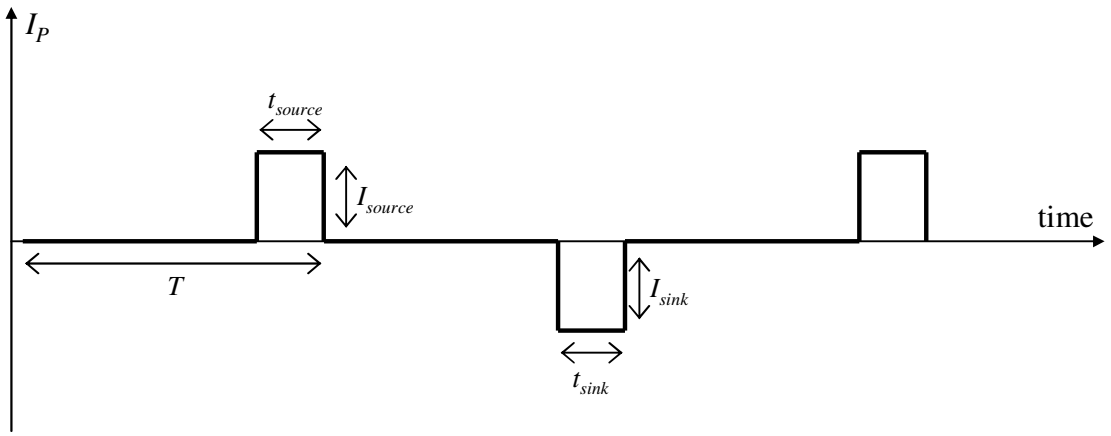

Figure 2.28 Charge Pump Output Pulses

Spurious tones are also generated by the fractional-N feedback divider component, these occur around the VCO output frequency at multiples of the fractional divider ratio *N*. When multiplied by *N*, spurious signal levels increase by *20Log(N)* [10], but their offset location from the reference frequency remains the same. For example: a

1 kHz spurious signal on a reference of 1 MHz when multiplied by N=1000 times will generate a 1 kHz offset spurious signal at the 1 GHz output frequency.

The abrupt change in phase due to N being changed on a periodic basis, as described earlier, will cause a spurious signal. This spur is called a fractional spur and can be located close to the frequency resolution, F_R/F, where F is the fractional modulus of the circuit, i.e. F = 8 would indicate a $1/8^{th}$ fractional resolution. It must be suppressed in some manner other than by the loop filter [1] and is much larger than the typical reference spur generated by an integer-N device. When in lock, the integer-N circuit's phase detector generates fast spikes that leak and modulate the VCO control line, generating spurs at the reference signal. Extra filtering reduces the spurious signal level, this is the main reason for the common use of third and fourth order loops.

The spurious output of the divider can be significantly reduced by using a sigma-delta modulator as a fractional-N divider. This method enables spurious signal reduction by over sampling. Over-sampling works by moving these undesirable components to higher frequencies and then filtering them out with the loop filter.

The occurrence of cycle slips when the DPLL is in lock is a highly non-linear phenomenon caused by large offsets of the phase error generally due to the noise sources mentioned above [37]. The cycle slip rate is thus dependent on the signal to noise ratio. Cycle slips occur when the V_C offset from the equilibrium instantaneously becomes large, due to noise. If the DPLL system is stable, then a cycle slip will not cause the DPLL to go unstable, but has the effect of reducing the immediate phase error and pushing the loop towards lock. After the first slip the control voltage V_C will have a wrong initial value, this is known as loop stress, and becomes more susceptible to another slip [37]. Thus for small damping factors cycle slips occur in bursts, clusters of cycle slips occur separated by long time intervals. With low noise levels the loop can only be pushed slightly away from the equilibrium and therefore in most cases cycle slips will not occur once the DPLL has locked.

In conclusion the DPLL loop is traditionally designed with a wide bandwidth so as to minimise the noise due to the VCO. The wide bandwidth also has the effect of

reducing the lock time of the loop; however the additional loop noise, due to the PFD, the loop filter, the reference source and the feedback divider components, will be passed by the loop filter. This contradiction is outlined in Figure 2.29 below.

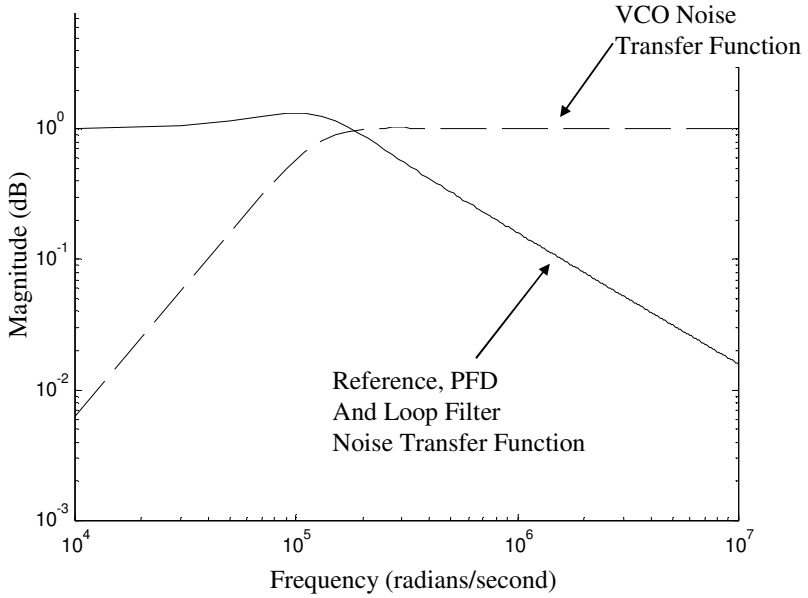

Figure 2.29 Transfer Function of Noise Sources

Having a loop filter with a sharp roll-off factor can help allay the trade off between both requirements. A sharp roll-off characteristic can best be achieved with a high order loop filter. Such high order systems are discussed briefly in the next subsection.

2.6 High Order Loops

The benefit of high order DPLL systems is that they can provide a significantly better attenuation of the loop noise relative to lower orders due to the sharper roll-off frequency characteristics of the high order systems. The correlation between increased filter order and increased noise attenuation of the loop is illustrated in Figure 2.30.

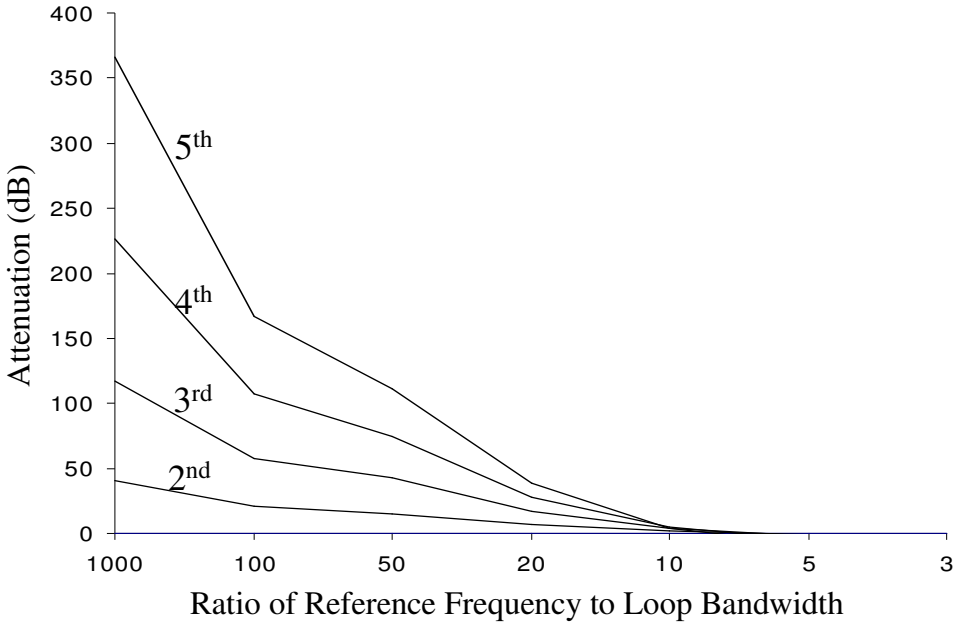

Figure 2.30 Plot of Increased Noise Attenuation as Order Increases

Looking at the system performance in the frequency domain, as given in Figure 2.31 below, it can be seen that as the order of the loop filter is increased the roll-off factor is increased, thus reducing the in-band noise.

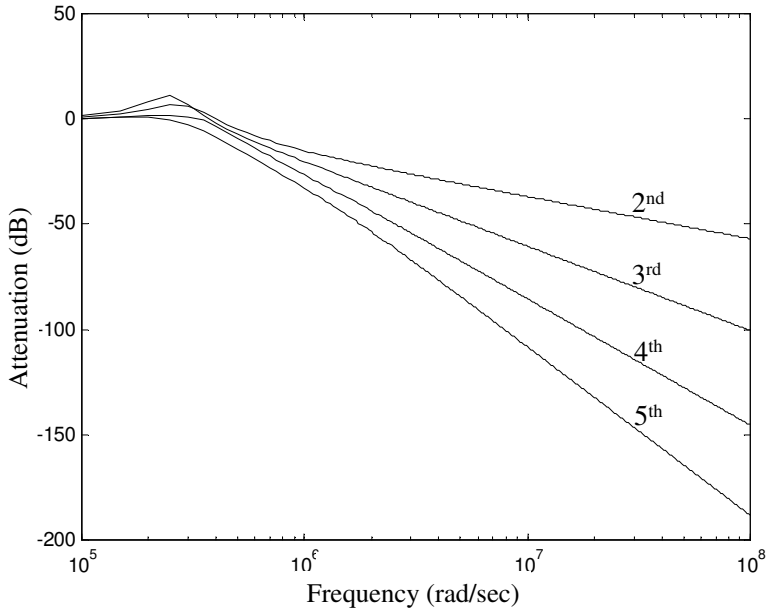

Figure 2.31 High Order Roll-off Factor

For maximum noise attenuation, it is desirable to have as sharp a roll-off factor as possible, it also needs to be considered whether additional poles or zeros will have a

desirable effect on the loop. For example it is common practice in the literature to include additional parasitic poles in the DPLL loop filter, [38]. This means that the designer can ignore the parasitic poles and use traditional second and third order criteria to place the dominant poles and design what is in theory a high order system [38]. The additional parasitic poles are used solely to attenuate the out-of-band noise and have minimal effect on the DPLL response [39]. However if all the system poles are placed in-band (the non-parasitic poles in Figure 2.16) then this can produce a sharp cut off of 20dB/decade for each system pole. However if this is the case then the phase margin is significantly degraded [39], stability issues ensue, and traditional design methods become ineffective. This is the major drawback with high order system design. Thus high order systems are considered risky and are rarely used in practice [13]. The lack of an accurate design method for higher order systems means that the design of such systems involves simulation alone. But simulation alone can be an arduous task due to the long simulation time, the chaotic nature of these systems [39], [40] and the limited knowledge of the system's stable regions.

Finally it is worth considering that the choice of loop order can have a greater impact on the system transients than would be expected. The loop filter always has at least one real pole, therefore even-order systems, for example the fourth order loop with four poles, has two real poles as shown in Figure 2.16 earlier, while odd order DPLL loops have only one real pole. The effects of this are that the odd order loops will place their system poles in a more efficient arc as will be illustrated in Chapter 3 when using a filter prototype, while the even order loops will have at least one real pole placed away from the arc of system poles with a less significant influence on the system response than would be expected. Since phase margin is small for odd orders, their advantage is a rather large attenuation of high normalised frequencies, on the other hand even order loops have a large phase margin and are advantageous for band-pass properties and additional attenuation.

2.7 Summary

The ideal DPLL system is both fast locking and low noise. This however is infeasible as both requirements are contradictory. The fundamental problem is that

both requirements are defined by the system bandwidth; a wide bandwidth is required for fast lock and low VCO noise but a narrow bandwidth for superior pre-filter noise characteristics. Traditionally a DPLL is designed by choosing a wide bandwidth where the loop filter bandwidth is kept sufficiently small, so as to minimise pre-filter noise. Increasing the order of the DPLL loop can further attenuate the pre-filter noise without necessarily narrowing the bandwidth.

It has been discussed in this chapter that increasing the order of the loop filter can further reduce the system noise. However designing such high order systems is an arduous task due to increased stability concerns and a lack of a valid design method. In this thesis a high order design methodology for DPLL systems using piecewise linear methods is proposed, this can be realised using a novel filter prototype methodology as a design tool for the DPLL, and a piecewise linear modelling methodology that exhibits increased accuracy relative to existing methods and can be extended to higher orders. This means that high order DPLL systems become less risky and realizable. The piecewise linear methodology is discussed in more detail in Chapter 5 of this thesis, while a novel high order design methodology is considered later in Chapter 6.

CHAPTER 3
DIGITAL PHASE LOCKED LOOP SIMULATION AND MODELLING

3.1 Introduction

System modelling is an important part of any design process; it assists in system analysis and the determination of the expected system performance either analytically or through simulation, by predicting the logical behaviour of the system. There are four methods of analysing and modelling the Digital Phase Locked Loop (DPLL): the linear PLL model; circuit level simulation; nonlinear methods; and behavioural modelling. Each of these techniques is assessed in detail in this chapter.

The most common method of DPLL analysis is to use a linearization of the DPLL to approximate the APLL response [14]. The linearization of the DPLL and its limitations will be considered in detail in the next subsection, the major assumption made is that the APLL and DPLL can be considered similar; this assumption will also be examined by comparing the nonlinear DPLL with the linear PLL. The second method of analysis is circuit level simulation; this considers the DPLL at the component level using software programs such as SPICE. As such simulations are at transistor level they are accurate but extremely slow relative to the analytical linear model [2]. It is therefore generally the case that this type of simulation is used only for the final verification of the design before the DPLL system is built [41], circuit level simulation is discussed briefly in Section 3.3. In Section 3.4 non-linear methods are discussed. In Section 3.5 behavioural models are considered, these consider the input-output behaviour of each component block in the DPLL loop. The major advantage of behavioural models is that they are accurate relative to the linear approximation [3] and fast compared to circuit level simulations. However existing methods are not amenable to high order system modelling because of the increased complexity of the model equations due to the existence of integral and differential terms.

3.2 Linear Phase Locked Loop Approximation

In this section the linear approximation of the APLL (also known as the linear PLL, LPLL) is considered in detail. The significant advantage of the LPLL is that it enables the designer to utilise powerful classical linear analysis tools. The derivation and limitations of this approximation will be considered in detail in this section.

The linear approximation is derived from the APLL with a mixer PD component. The assumption that the LPLL can be used to model the DPLL will be considered later in this section. The linearization is made by first considering the mixer phase detector component. This nonlinear component multiplies two signals together as shown in Figure 3.1. It is possible to linearise it by making the following set of assumptions.

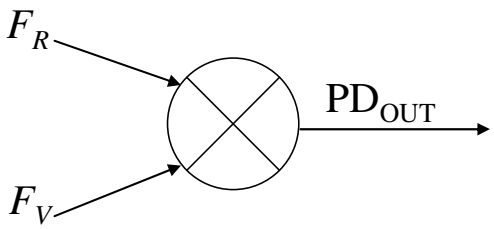

Figure 3.1 APLL Multiplier PD

The sinusoidal inputs of the APLL can be described as in equations (3.1) and (3.2) below.

$$F_R = E_R sin(\omega_R t + \theta_R) \quad (3.1)$$

$$F_V = E_V cos(\omega_V t + \theta_V) \quad (3.2)$$

where E_R and E_V are the amplitude of the reference and VCO output signals respectively, and θ_R and θ_V are the phase of the reference and the VCO output signals respectively. The output of the mixer PD is calculated by multiplying both input signals together, as in equation (3.3).

$$PD_{OUT} = K_P(Sin((\omega_R - \omega_V)t + \theta_R - \theta_V) + Sin((\omega_R + \omega_V)t + \theta_R + \theta_V)) \quad (3.3)$$

where K_P is the phase detector gain. This can be linearised as follows:

1. Assume that the low pass loop filter will attenuate the high frequency component $sin((\omega_R + \omega_V)t + \theta_R + \theta_V)$.
2. For a small phase offset the reference and VCO frequencies are approximately equal ($\omega_R \approx \omega_V$), therefore the $sin((\omega_R - \omega_V)t + \theta_R - \theta_V)$ component is approximately equal to $sin(\Delta\phi)$, where $\Delta\phi$ is the phase difference between the reference and VCO output signals, θ_R and θ_V respectively.
3. Finally for a small slowly varying signal close to lock, the remaining $sin(\Delta\phi)$ component can be approximated by $\Delta\phi$.

With these assumptions the PD output can be approximated to the linear equivalent of:

$$PD_{OUT} = K_P\Delta\phi \tag{3.4}$$

This can be represented graphically by the linear PD subtractor component of Figure 3.2.

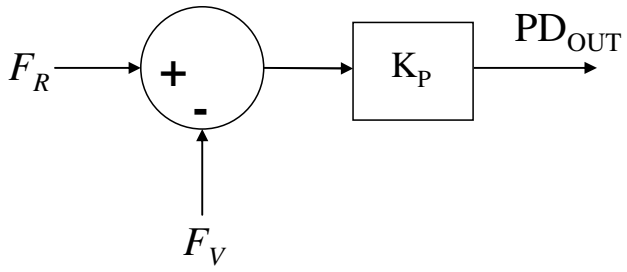

Figure 3.2 Linearised PFD

This approximation is sometimes referred to as the continuous time approximation (CTA) and is only valid when the loop bandwidth is small relative to the reference frequency [2], or more specifically no greater than one tenth of the reference frequency [19].

To complete the linearization of the overall APLL loop the VCO component block also needs to be considered. The VCO component generates an oscillating signal whose frequency is related to the input control voltage V_C. Ideally the output frequency of the VCO will increase linearly as the VCO input voltage is increased.

In reality this is not the case, the output frequency saturates as the VCO input voltage exceeds some threshold value, as shown in Figure 3.3.

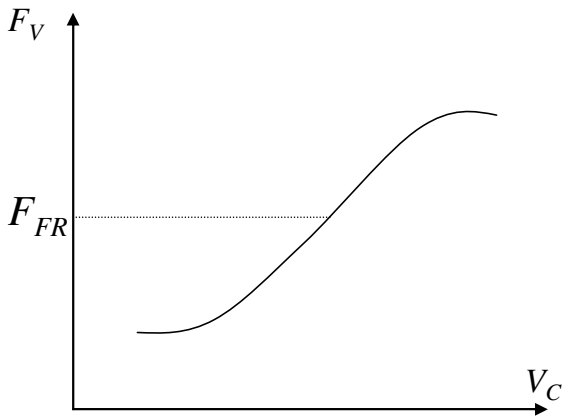

Figure 3.3 VCO Non-linearity

This saturation means that the VCO input to output relationship is non-linear. The VCO can be assumed linear if the DPLL system operates away from these saturation regions, i.e. between V_{C_min} and V_{C_max} in Figure 3.4.

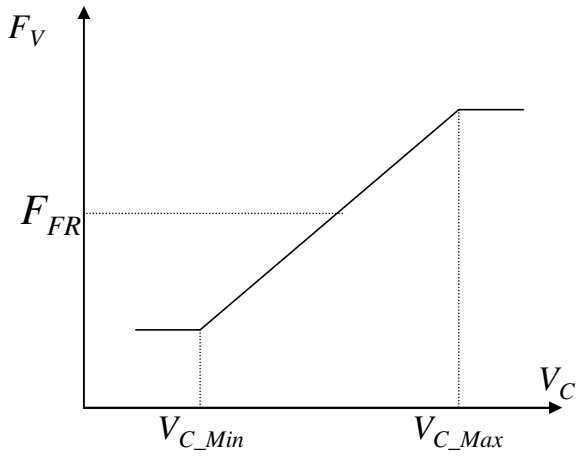

Figure 3.4 VCO Linear Approximation

Making this assumption the VCO can be approximated to a gain and an integrator component where the output frequency corresponds to:

$$F_V = \frac{K_V V_C}{s} \tag{3.5}$$

where K_V is the VCO gain in radian/s/V. Using the expression in equation (3.5) the VCO can be modelled linearly, as illustrated in the block diagram of Figure 3.5.

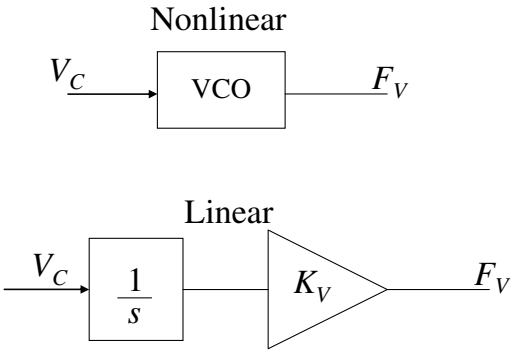

Figure 3.5 Linear Equivalent of the VCO

Replacing the nonlinear PD and VCO components with the linear approximated subtractor, integrator and gain components as discussed earlier, the APLL can be approximated by the LPLL block diagram of Figure 3.6.

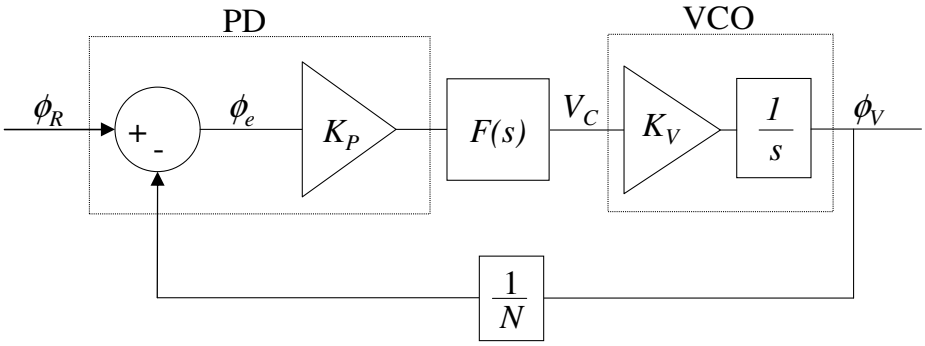

Figure 3.6 Linear PLL System Block Diagram

The linear approximation is found to be reasonable when characterising the PLL close to lock. In this situation the system state changes by only a small amount on each cycle of the reference signal. This assumes that the detailed behaviour of the loop within each cycle is not important and only the average behaviour over many cycles is important. By applying an averaged analysis, the time-varying operation can be bypassed and the linear approximation can be applied [19, 42].

Using the LPLL block diagram of Figure 3.6, the closed loop system transfer function is calculated as that of equation (3.6), where K_P is the charge pump gain in amps, K_V is the VCO gain in radians/second/volt and $F(s)$ is the loop filter transfer function. Similarly the loop error transfer function for this system can be calculated and is given by equation (3.7).

$$H(s) = \frac{K_P K_V N F(s)}{Ns + K_P K_V F(s)} \tag{3.6}$$

$$H_e(s) = \frac{Ns}{Ns + K_V K_P F(s)} \tag{3.7}$$

In the literature the most common PLL considered is the APLL, as it is generally accepted that the DPLL can be represented accurately by using the linearised APLL model [27]. Both the DPLL and APLL are considered similar because they both have the same information at the critical nodes of each loop, i.e. frequency at the PD input, and phase error at the loop filter output. This assumption will be considered in detail later in Section 3.2.1. The major difference between the APLL and the DPLL is the switching quantization-like nature of the CP-PFD; this non-linearity is inherent to the operation of the loop and cannot be ignored if an accurate model is required. Because of this the linearization is not entirely accurate [2] and does not fully capture the complexity of DPLL behaviour. Thus it is generally the case that empirical design and simulation are used concurrently to ensure the correct transient behaviour. This design method is outlined graphically in the flow chart of Figure 3.7.

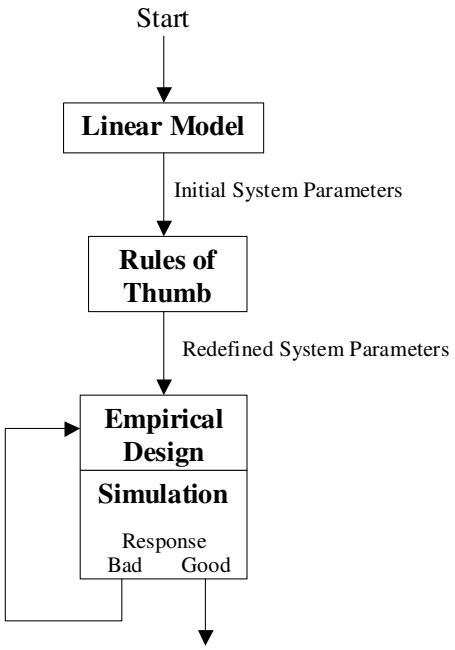

Figure 3.7 Traditional PLL Design Process

Traditionally the designer starts with the linear model, applies rule-of-thumb guides to choose system parameter values. The system transients are then approximated

using the linear model. Empirical design methods are then used to redefine these parameter values until the desired linear transients are produced. The final step of the design process is to use a circuit level simulation to verify the expected performance of the DPLL. If the results of the circuit level simulation are not satisfactory, the parameter values are reselected and the process starts again, until eventually the DPLL performs as desired at circuit level. This design methodology can be extremely slow depending on the accuracy of the initial linear model and more importantly the experience of the designer in preselecting and reselecting the parameter values. The circuit level simulation is significantly slower than the analytical linear model and thus is only used as the final stage of the design process to verify the expected PLL performance.

3.2.1 Linear and Digital Phase Locked Loop Comparison

As discussed earlier it is common practice to use the LPLL system to approximate the DPLL, the significant difference between the two being the operation of the respective phase detectors. The LPLL PD is a linear adder component whilst the DPLL PFD is a more complex non-linear state driven device. In this section both systems are compared, concentrating initially on the PD blocks and then looking at the relative performance of both closed loop feedback systems. The specific digital PFD component that is considered is the CP-PFD as described earlier.

A comparison of the output of the linear PD and CP-PFD for a phase offset of $\pi/2$ can be achieved by considering the output of both blocks to two typical input signals. However it must first be considered that both components are operating on different signal types, i.e. the CP-PFD inputs digital signals, while the linear PD has sinusoidal signals at its inputs, as shown in the top two plots of Figure 3.8. The PD output of both systems is shown in the bottom plot of Figure 3.8. A simple comparison can be made between the two systems by calculating the phase offset between the two input signals, as plotted in Figure 3.9.

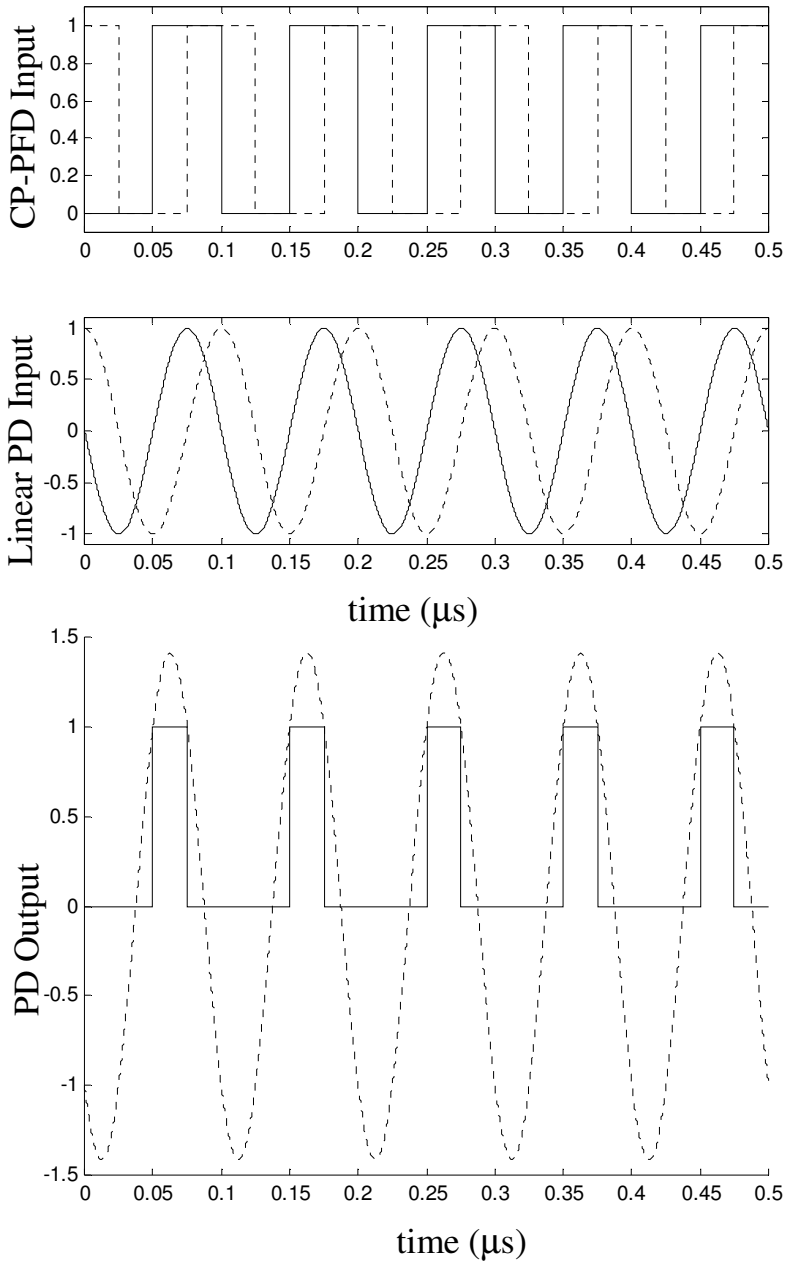

Figure 3.8 Plot of PD Inputs (feedback – dashed line; reference – continuous) for the LPLL and DPLL, and the DPLL CP-PFD output (continuous), and the LPLL PD output (dashed) for a small phase offset.

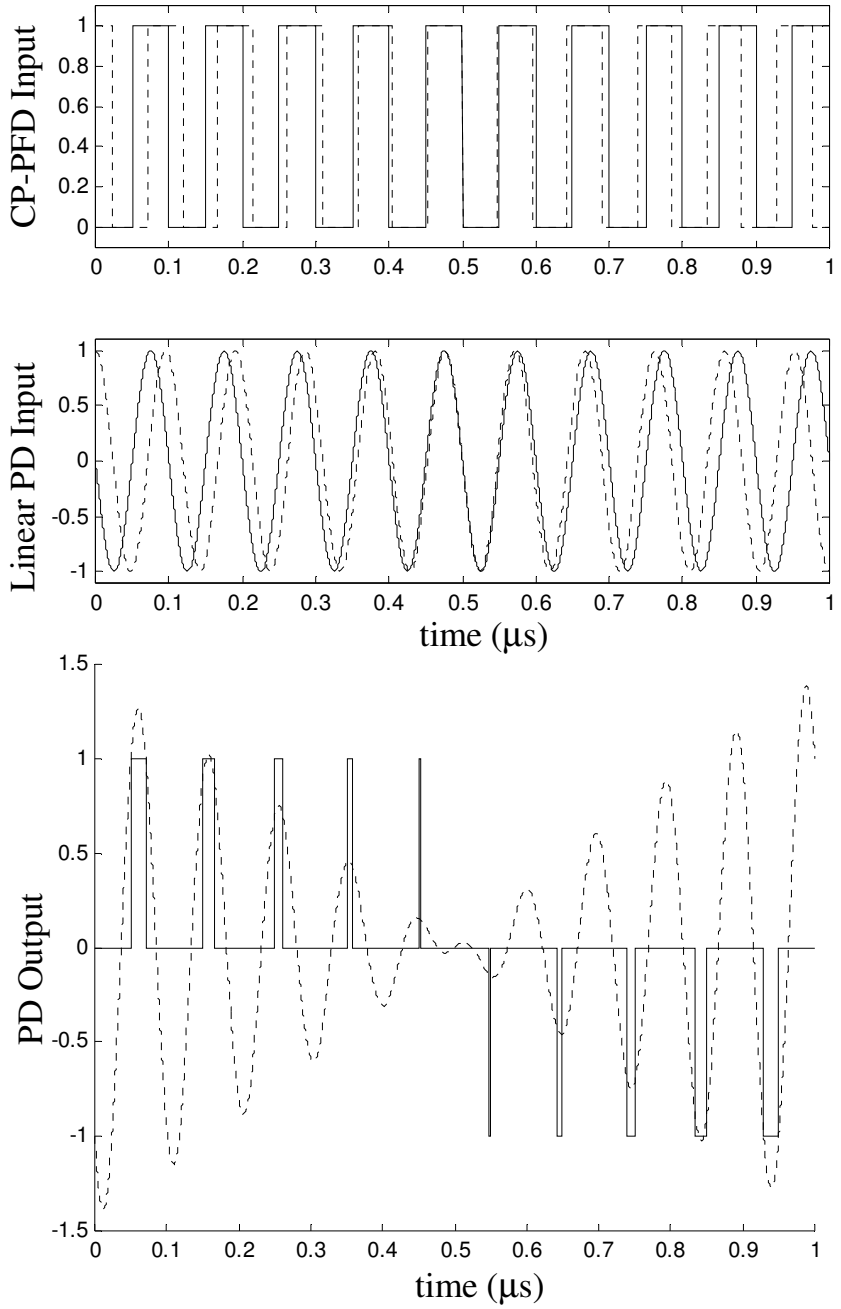

Figure 3.9 Plot of PD Inputs (feedback – dashed line; reference – continuous) for the LPLL and DPLL, and the DPLL CP-PFD output (continuous), and the LPLL PD output (dashed) for a small frequency offset.

As both systems approach zero phase offset at time 0.5μs, both outputs approach zero, when the reference signal (continuous line) leads the VCO signal (dashed line) the PD outputs are some positive value. Likewise, as the VCO begins to lead the reference (after 0.5μs) on average both PD outputs become increasingly negative.

If one considers the average phase error over each cycle for both systems it can be seen that as the phase error increases, the difference between the linear and digital systems becomes more pronounced, as illustrated in Figure 3.10. This verifies the linear approximation requirement discussed earlier, illustrating that the approximation is only valid for a small phase error, i.e. when the loop is close to lock.

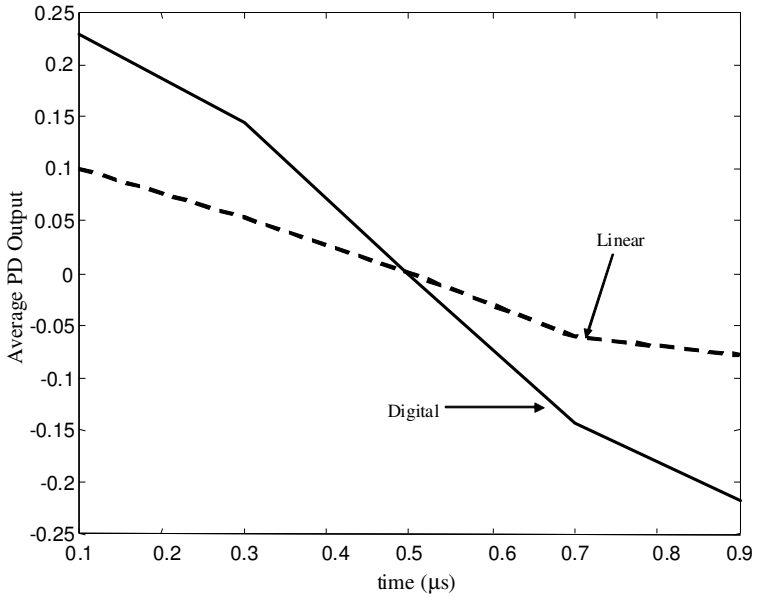

Figure 3.10 Average Error over each Cycle of the Reference Signal

The second requirement of the linear model is that of the continuous time approximation, this states that the loop filter bandwidth should be chosen to be no greater than $1/10^{th}$ of the reference frequency. In Figure 3.11 a plot of the filtered output of the PD is shown, the filter bandwidth is chosen to be $1/10^{th}$, $1/5^{th}$ and ½ that of the reference frequency.

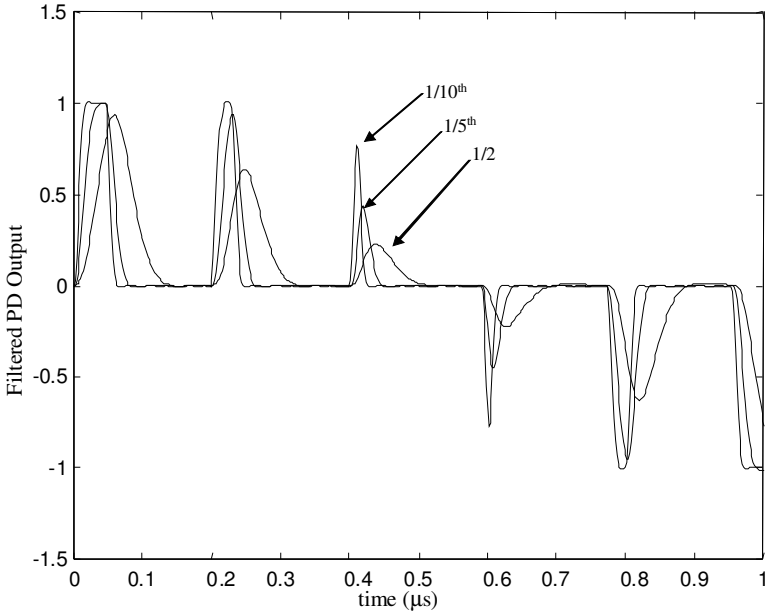

Figure 3.11 Filtered Outputs of both the Linear and Digital PLL

From Figure 3.11 it can be seen that as the loop filter bandwidth is reduced and brought closer to the reference frequency, the filtered CP-PFD output begins to degrade. This degradation of the CP-PFD output removes important frequency components from the control voltage signal, which are necessary to drive the VCO. Therefore for the continuous time approximation to remain valid it is prudent design to choose a loop bandwidth no greater than $1/10^{th}$ that of the reference signal.

From the above analysis it is reasonable to assume that the linear PD is a good approximation of the digital CP-PFD when the phase error is small and when the system bandwidth is much less than the reference frequency. In terms of the overall closed loop LPLL and DPLL systems, they can be compared by looking at the transient response of both loops to a step in the reference phase ϕ_R, and a step in the reference frequency F_R.

With the LPLL the phase error is a continuous time representation of the phase difference between the reference and VCO signals. The CP-PFD output signal shown earlier in Figure 3.8 is not a continuous time signal, therefore in order to compare the linear and digital PLL a continuous time representation of the above DPLL phase difference must be found. To find the continuous time phase error

between the reference and VCO signals of the DPLL, consider the two square wave signals in Figure 3.12, the top one representing the reference and the bottom one representing the VCO or feedback signal.

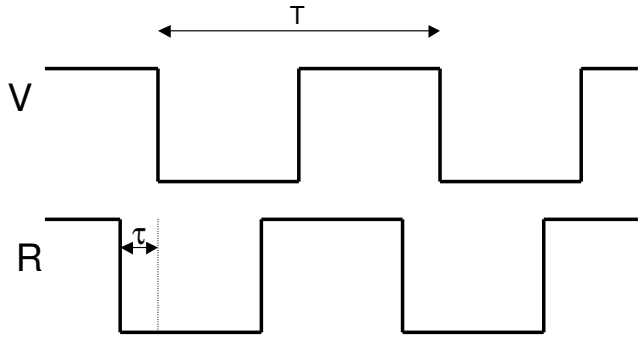

Figure 3.12 CP-PFD Input Signals

The frequency of the reference and VCO signals are equal to *1/T*. If we consider the falling edges of both signals, then the time difference from the falling edge of one signal to the other is τ. The continuous time phase difference between these two signals is equal to:

$$\phi_e = \frac{2\pi\tau}{T} \text{ radians/s} \tag{3.8}$$

From the system transfer function given earlier in (3.6) the transient response of this linear transfer function should ideally equate to that of the DPLL phase error response. Using linear theory the phase and frequency step input responses of the LPLL are estimated as follows.

For a phase step the LPLL phase error is calculated as:

$$\phi_e = H_e(s)\frac{\Delta\phi}{s} \tag{3.9}$$

where $\Delta\phi$ is the change in phase, and $H_e(s)$ is as given in equation (3.7). Therefore the phase step response of the LPLL is:

$$\phi_e(s) = \frac{\Delta\phi Ns}{Ns^2 + K_V K_P F(s)s} \tag{3.10}$$

where *F(s)* is the loop filter transfer function as given in Table 2.2 earlier. Similarly the phase error response of the LPLL for a frequency step of $\Delta\omega$ is:

$$\phi_e(s) = \frac{\Delta\omega N s}{N s^3 + K_V K_P F(s) s^2} \tag{3.11}$$

The step response of the DPLL is determined using a circuit level simulation; this is achieved by locking the DPLL and then varying the input reference phase or frequency with a step in the input signal. Consider the following example, the LPLL and DPLL are modelled and simulated respectively and the response of both systems are compared, the system component values for this example are filter capacitor C_2 = 20pF, filter resistor R_2 = 9kΩ, VCO gain K_V = 1 MHz/V, and charge pump gain K_P = 10^{-6}A/(2π rad). In Figure 3.13 a plot of the DPLL and LPLL phase step responses for a phase step of π is shown.

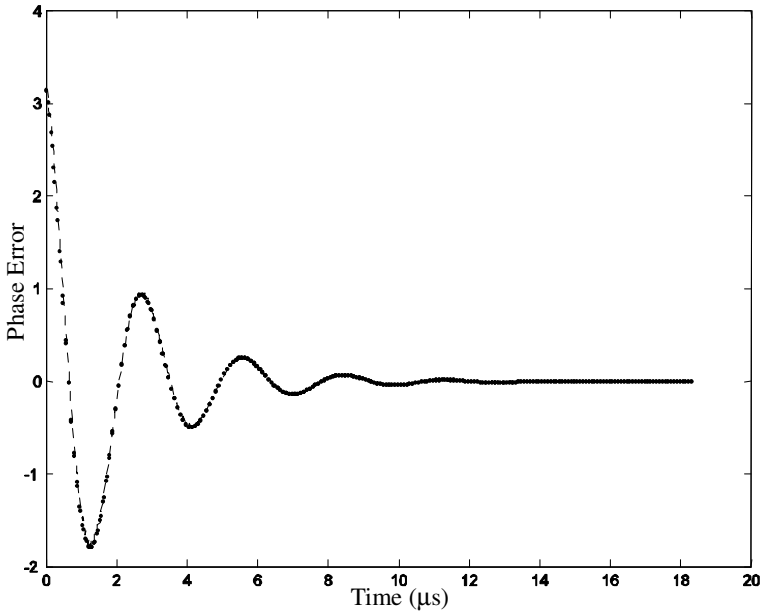

Figure 3.13 Phase Error Responses of LPLL(--) and DPLL(..)

It can be seen that the difference between the two is inconsequential. However as is expected the two phase error responses do not match exactly, this difference is plotted in Figure 3.14.

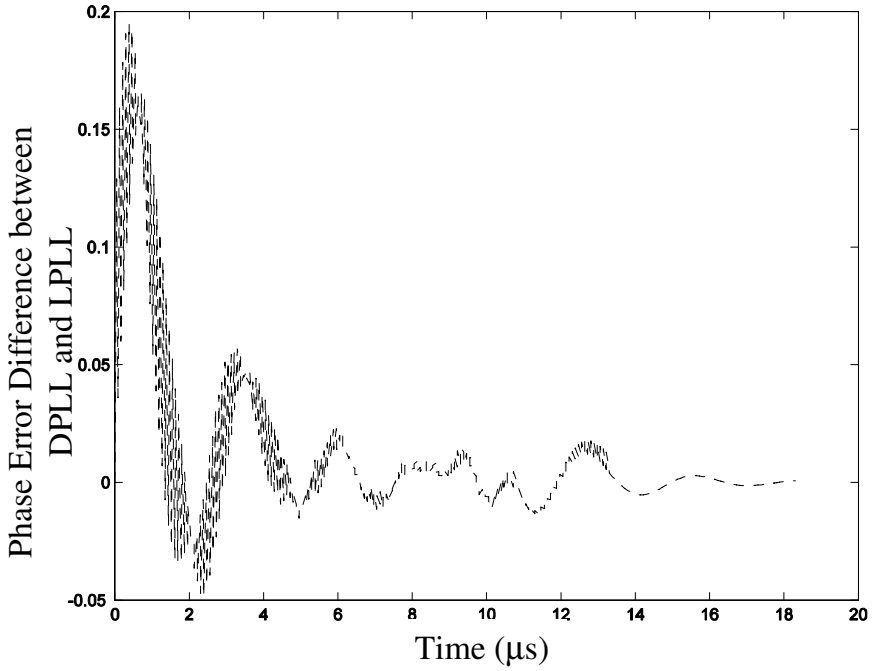

Figure 3.14 LPLL and DPLL Phase Error Difference

The difference in the phase error responses of both systems are minimised as both loops approach the locked state. The non-linear nature of the DPLL loop results in a difference between the linear and nonlinear responses, this difference is due to the inaccuracy in the linear approximation and because the non-linearity in the CP-PFD is minimal in the locked state but becomes more pronounced when the DPLL is out of lock. The comparison is carried out for a range of phase step sizes, Figure 3.15 shows the difference between both system responses for the various phase step inputs from $\pi/12$ to π. As expected the difference between the LPLL and DPLL increases as the phase step is increased.

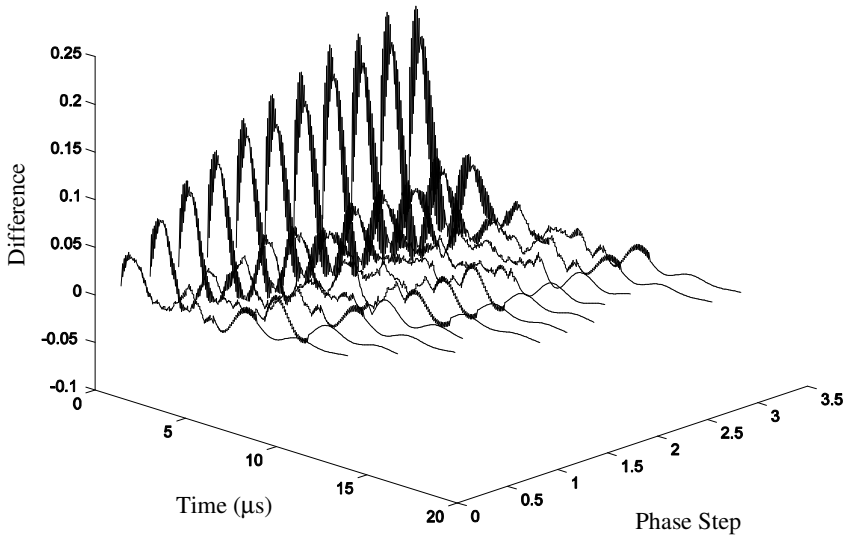

Figure 3.15 Linear and Digital PLL Difference for a Range of Phase Steps

The difference between the DPLL and LPLL phase error response increases as the phase step size is increased. So the DPLL is more linear if it operates within a small region, close to lock and does not deviate outside this region. In the case of a frequency step input we expect to find similar results, i.e. an increase in difference between the phase errors of both systems as the input frequency step size is increased. This is shown in Figure 3.16.

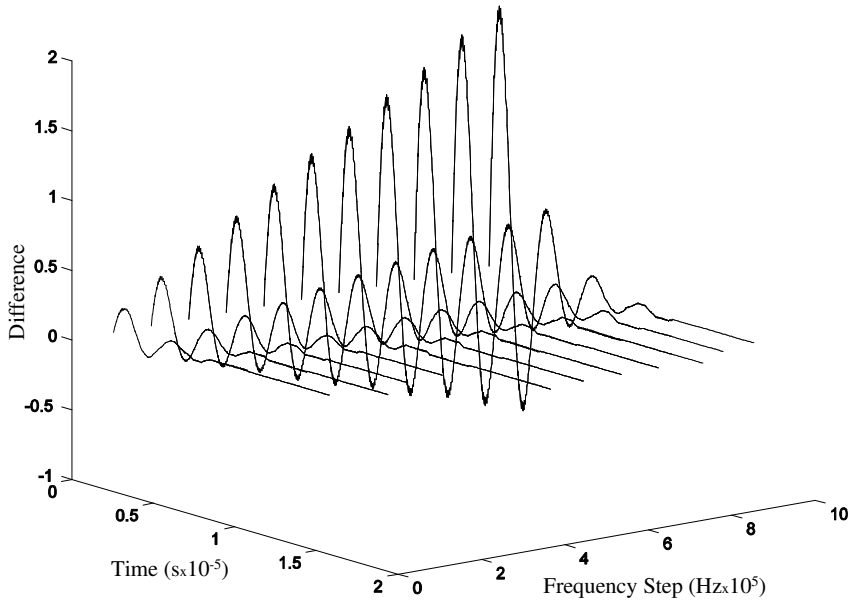

Figure 3.16 Linear and Digital Difference for a Range of Frequency Steps

3.2.2 Linear Stability Boundary

One of the simplest and most utilised linear analysis tools is the stability boundary of the DPLL system. Once the boundary is determined for a particular system, it is used as the starting point of the design methodology outlined in Figure 3.7 earlier. An example of a second order linear stability boundary is the linear boundary defined by Gardner [27], this identifies the stability boundary using the system transfer function of equation (3.6) and is calculated as shown in equation (3.12) below.

$$K' = \frac{1}{\frac{\pi}{\omega_R \tau_2}\left(1+\frac{\pi}{\omega_R \tau_2}\right)} \tag{3.12}$$

where K' is equal to $K\tau_2$, and where K is the system gain $K_P K_V$ and τ_2 is the second order DPLL loop filter time constant. Utilising equation (3.12) and choosing a filter time constant τ_2, and a range of reference frequencies (ω_R radians/second), a plot of the linear stability boundary for this particular system can be determined. This boundary is shown in Figure 3.17.

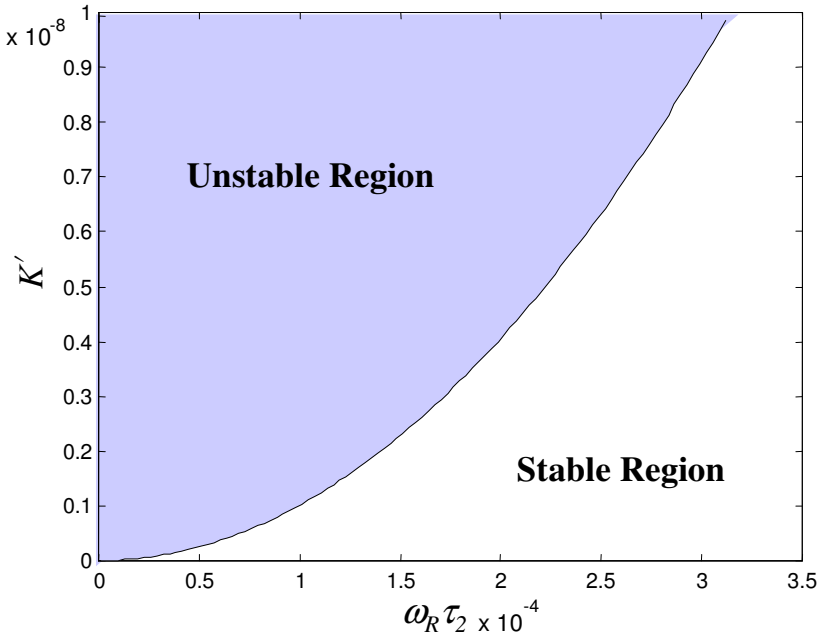

Figure 3.17 Second Order Linear Stability Boundary

With this plot it can be seen that there are a range of values of the system gain parameter (K') and of $\omega_R\tau_2$ that are stable, by choosing parameters within this region it is expected that a stable DPLL system can be designed. Similarly for the third order system Gardner [27] offers a stability boundary as defined by (3.13), where, $b = 1+(C_2/C_3)$ and $a = exp(-2\pi b/\omega_R\tau_2)$.

$$K' < \frac{4(1+a)}{\frac{2\pi(b-1)}{b\omega_R\tau_2}\left[\frac{2\pi(1+a)}{\omega_R\tau_2}+\frac{2(1-a)(b-1)}{b}\right]} \tag{3.13}$$

Finally consider, as an example, Gardner's linear stability criterion, this linear boundary is plotted again in Figure 3.18 for a particular set of DPLL system parameters. The parameters chosen in this example are F_R=12MHz, R_2 = 10kΩ, C_2 = 32.5nF, K_P = 1mA and K_V = 160 Hz/V.

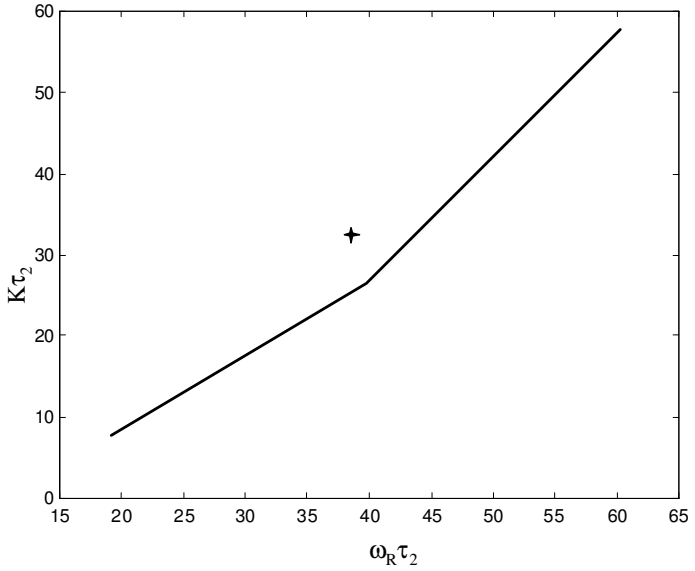

Figure 3.18 Linear Stability Limit

The line represents the stable limit for this particular system, while the star represents the chosen set of parameter values for the particular DPLL system. Gardner states that in order to be stable the system must be below this line, so accordingly this particular system should be unstable. However the actual response of this system as plotted in Figure 3.19 shows that the system is in fact stable.

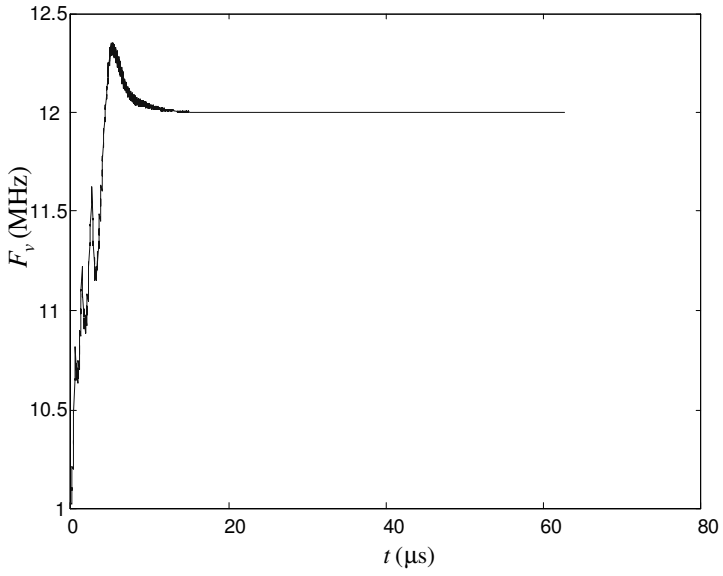

Figure 3.19 System Response

This particular example illustrates the known inaccuracies when applying linear theory to the DPLL. However despite the inaccuracies of the linear methods, they do provide a starting point from which empirical design methods are used to eventually choose the optimum system parameter values. These design methods are generally complemented with a circuit level simulation, to ensure that the DPLL operates as expected. Alternative design techniques to those described above include circuit level simulation, nonlinear methods and behavioural methods, each of these are discussed in detail later in this chapter.

3.2.3 Existing Linear DPLL Design Methodologies

In this section alternative DPLL design methodologies that are derived from the LPLL are considered. In contrast to the linear stability criteria of the previous section the methods discussed in this section do not provide a global prediction of the DPLL stability boundary. Instead they offer estimates of specific system parameter values for a chosen loop performance, such as the settling time [26], the undamped natural frequency and damping factor [43] or the phase margin [13].

The first criterion to be considered is that of Mirabbasi [26]. This method uses the Bessel filter prototype to determine the system parameter values for the third order DPLL. The component values are defined as in equations (3.14-3.18) given below.

$$\omega_C = \frac{2.5}{t_s} \tag{3.14}$$

$$B_L = 1.05\omega_C \tag{3.15}$$

$$C_2 = \frac{K_P K_V}{3\omega_C^2} \tag{3.16}$$

$$C_3 = \frac{C_2}{5} \tag{3.17}$$

$$R_2 = \frac{1}{C_2 \omega_C} \tag{3.18}$$

Using the parameter values as calculated above, the designer expects to produce a DPLL system with a settling time of t_s and a noise bandlimit of B_L. In general though, this is not the case and so the parameter values are varied slightly until the correct system performance is verified using a circuit level simulation.

The second design method derived by Williamson [43] determines the parameter values of a stable fourth order DPLL from a predefined choice of undamped natural frequency, ω_n, and the damping factor, ζ, using the set of equations given in (3.19-3.23) below.

$$C_2 = \frac{K_V I_P}{\omega_n^2 N} \tag{3.19}$$

$$R_2 = 2\zeta \sqrt{\frac{N}{K_V I_P C_2}} \tag{3.20}$$

$$C_3 = \frac{C_2}{12} \tag{3.21}$$

$$R_4 = 3R_2 \tag{3.22}$$

$$C_4 = \frac{R_2 C_2}{20 R_4} \tag{3.23}$$

For best performance it is suggested that the closed loop system phase margin, ϕ_p, should be chosen between 30° and 70°, but generally should be chosen to be 45° [10]. In the above set of equations ω_n is chosen to be $5\pi/t_{LCK}$, and for an optimum phase margin of 45°, ζ is chosen to be equal to 0.707.

The final design methodology as proposed by O'Keese [13] also considers the phase margin, The filter component values of this fourth order DPLL are determined from (3.24-3.30) below. O'Keese also suggests, as a rule of thumb, that the filter capacitor C_4 should be chosen no greater than one tenth of C_3, this insures that the additional pole will add a phase shift of no more than 4-5º [10]. If this rule is not adhered to the pole due to C_4 will interact with the dominant system poles possibly causing instability. This requirement was discussed earlier in Section 2.4, where additional high order poles are recommended to be placed out-of-band so that they do not

interact with the PLL systems dominant poles, thus second and third order stability methods can be used to design stable high order systems.

$$T_3 = \frac{\sec(\phi_p) - \tan(\phi_p)}{\omega_n} \tag{3.24}$$

$$T_4 = \sqrt{\frac{10^{\left(\frac{ATTN}{10}\right)} - 1}{(2\pi \cdot F_R)^2}} \tag{3.25}$$

$$\omega_n = \frac{\tan(\phi_p) \cdot (T_3 + T_4)}{(T_3 + T_4)^2 + T_3 T_4} \cdot \left[\sqrt{1 + \frac{(T_3 + T_4)^2 + T_3 T_4}{\left[\tan(\phi_p) \cdot (T_3 + T_4)\right]^2}} - 1 \right] \tag{3.26}$$

$$T_2 = \frac{1}{\omega_n^2 (T_3 + T_4)} \tag{3.27}$$

$$C_3 = \frac{T_3}{T_2} \frac{K_P K_V}{\omega_n^2 N} \sqrt{\frac{1 + \omega_n^2 T_2^2}{(1 + \omega_n^2 T_3^2)(1 + \omega_n^2 T_4^2)}} \tag{3.28}$$

$$C_2 = C_3 \left(\frac{T_2}{T_3} - 1 \right) \tag{3.29}$$

$$R_2 = \frac{T_2}{C_2} \tag{3.30}$$

Each of the design methodologies outlined above are found to assign parameter values that do produce stable DPLL response, as long as certain rules of thumb are utilised. However, similar to Gardner's model, all of the above methods do not guarantee the system stability; this is to be expected as all of these models are derived from the same linear transfer function.

3.2.4 Discrete Z-Domain Analysis of the Digital Phase Locked Loop

To complete this section on linear modelling methods it is necessary to consider the discrete z-domain analysis methodology. This method is a discrete sampling domain analysis approach. Considering the discrete nature of the DPLL the z-domain is expected to offer a more accurate approach to the analysis of the discrete system compared to the continuous linear approach considered thus far [44-46]. This increased accuracy is at the expense of an increased model complexity.

The z-Domain model is determined from the continuous linear transfer function given earlier in equation (3.6) using the bilinear transform expression given in equation (3.31) below.

$$H(z) = H(s)\Big|_{s=\frac{2(z-1)}{T(z+1)}} \tag{3.31}$$

The main advantage of the bilinear transformation over other transformation methods is that it maintains the magnitude characteristics of the original transfer function [47]. The system transfer function determined from this transformation is used as an alternative to equation (3.6) to model the DPLL and in particular the discrete nature of the DPLL system.

This z-domain model is not commonly used in practice because it exhibits the same drawbacks as the continuous method. This is because it is derived from the linear approximation of the APLL (using continuous time assumptions) thus removing the most significant discrete components of the loop. Therefore it cannot accurately model the discrete nature of the DPLL, but only the discrete nature of the linearised PLL model. It can be concluded that the z-domain model does not remove the inaccuracies that exist in the case of the continuous linear model. Also in the case of high order DPLL systems it has been concluded in [27, 44] that the z-domain model becomes incrementally more cumbersome as the system order increases. For all of these reasons the z-domain model is not a viable alternative to the continuous model. In the best case scenario it offers only a slight improvement in accuracy for low order DPLLs relative to the continuous model, but with an increase in the model complexity.

3.3 Circuit Level Simulation

Circuit level simulators are a fundamental part of the design process of highly complex or mixed signal systems where an accurate analytical or linear system model does not exist. It is crucial for the designer to be confident of system performance before the expense of building the device is carried out. In particular it is necessary to determine the effects of noise on the system performance by adding models of all possible DPLL loop noise sources. Such noise sources can best be modelled using a circuit level simulator [41]. However using a circuit level simulator such as SPICE results in sampling at the highest frequency; this is at a rate of 5-6 times the Nyquist rate [48], thus many signals within the system will be excessively oversampled. As the simulation accuracy is indirectly proportional to the simulation sample time, it is well documented that circuit level simulators are up to four orders of magnitude slower than equivalent behavioural models [2]. For example a full SPICE circuit simulation of a DPLL for 20 μs response, requires up to 24 hours of CPU processing time, a major drawback.

A typical example of a mixed signal system is the DPLL which is a highly nonlinear and complex system. Despite this there is a large body of work on the linear analysis of DPLL systems. However linear methods are generally employed in conjunction with circuit level simulation, as outlined earlier in Figure 3.7. This is essentially due to the fact that the exclusive use of circuit level simulation is inefficient in terms of time, while at the same time the exclusive use of linear methods cannot guarantee the accuracy of the analysis [3]. A solution to this lengthy empirical design process is to replace the relatively inaccurate linear model with behavioural models [3, 13, 26, 49, 50]; these are fast and accurate, with the benefit of considerably reducing the design time. The existing behavioural models are considered in Section 3.5.

There is a large body of work on the circuit level design of each component block of the DPLL however a good reference and starting point is given in Razavi [15]. The exact design and performance of a DPLL circuit level model is beyond the scope of this thesis; however throughout this thesis circuit level simulations, along with other

published behavioural and event driven DPLL models, have been used as a reference and guide to ensure the accuracy of the results attained in this thesis.

3.4 Nonlinear PLL Design Methods

Traditional linear design methodologies as discussed in Section 3.2 are preferred due to the simple, powerful and advanced mathematical analysis. However as the DPLL is a nonlinear device, applying nonlinear analysis methods to the DPLL system will result in increased accuracy of the model, thus increasing the speed of the design process. While a large number of different nonlinear modelling techniques exist; these have only rarely been applied to the analysis of PLL systems [12, 42, 51-53]. This is because these techniques, by their very nature, complicate the design of an already complex system. In most cases they do not offer any practical solution to the design of high order DPLLs. This subsection will briefly review the existing nonlinear methods that have been applied to the DPLL in the literature. One such piecewise linear method, behavioural modelling, will be considered in the next subsection and will be the basis for the nonlinear design methodology proposed in this thesis.

The most commonly applied nonlinear analysis methodologies are Lyapunov's methods. These methods allow the stability of a system to be verified without having to explicitly solve the system equations. There are two methods of stability analysis introduced by Lyapunov: the indirect and direct methods. The indirect method states that if a nonlinear system operates within the vicinity of an equilibrium point, then the stability properties of that system are the same as its linearised approximation (note that this is the same assumption that was made earlier to validate the linearization of the APLL). The direct method on the other hand is a more powerful nonlinear analysis tool; it states that if the total energy in a system is decreasing for all time, then this system is globally asymptotically stable. The direct method of Lyapunov is the most general approach to the theory of stability of dynamical systems [54], and is considered here. The basic idea of the second method is to construct a scalar energy-like function of the system (a Lyapunov function) and assess whether it decreases over time. If it decreases then this suggests that the PLL

system is a stable system. The major disadvantage of Lyapunov's direct method is the need to find a suitable Lyapunov function for a particular system. If no Lyapunov function can be found for this system then no conclusions can be drawn about stability. This is a complicated task even for the simplest of systems. The problem becomes even more complicated as the DPLL order increases. This is due to the exponentially increasing complexity – thus finding a Lyapunov function becomes very challenging. Both Abramovitch [14] and Wu [53] concede that the Lyapunov method is not suitable for high order PLL system design (i.e. greater than second order). However a low order implementation of Lyapunov's direct method, known as Lyapunov redesign, has been applied in the literature to the design of second and third order APLL systems in [42], and [52].

The difficulty with applying Lyapunov and Lyapunov derived methods to high order PLL systems has led to the exploration of other nonlinear analysis tools suitable for numerical techniques. Abramovitch [12] and Wu [53] state that the Circle and Popov Criteria may be a viable alternative to Lyapunov. They are derived from two particular types of Lyapunov function candidates, shown below respectively[iv]:

$$V(x) = x^T Px \tag{3.32}$$

$$V(x) = x^T Px + 2\eta \int_0^y \psi(\sigma)(1-\alpha)d\eta \tag{3.33}$$

The methodology used is to combine one of these criteria with a search over a prescribed set in the filter parameter space. The standard approach is to use the Circle Criterion first and if this fails to use the Popov Criterion. In order for this approach to work a computationally expensive simulation is required, for this reason

[iv] **Note:** The Popov Criteria is the same as the circle Criterion when $\eta=0$

the circle criterion is not used for PLL design. A more detailed application of this nonlinear methodology to the PLL is given in [53].

LaSalle's theorem [55] is another extension of the Lyapunov method. The major difference between LaSalle's theorem and theorems such as Lyapunov, Circle, or Popov is that while these theorems extend to time-varying systems, LaSalle's theorem requires time-invariance. The PLL is a time varying system, however when in a locked state the system can be viewed as time invariant. This again is similar to the assumption of time invariance made during the linearization of the DPLL. Considering that the same assumption is being made in both cases and that the linear analysis is a simpler and more powerful tool it is understandable that LaSalles nonlinear method has not been applied to the DPLL in the literature.

As an alternative Rantzer [51] introduces another method similar to Lyapunov's direct criterion, known as the "almost global stability criterion" (AGSC). The basic difference between the Lyapunov method and the AGSC is that the latter uses a weaker notion of stability. For example in [51] an example of a system that is not globally stable is given, and therefore not stable in the sense of Lyapunov. The AGSC is applied and receives results that reflect the 'almost global stability' nature of that system. This concept of 'weakness' is achieved by using a 'density function' that is derived from a Lyapunov function. This density function may exist even if the system is not globally asymptotically stable, in which case the system can be said to have almost global stability. The problem again however is the difficulty in finding a unique density function for the system – this is not a simple task.

A common theme is evident in all of the nonlinear methods mentioned here; this is the complexity of the analysis, in particular with the application of such methods to the DPLL system. The DPLL with a CP-PFD is a highly complex system, attempting to apply complex nonlinear methods to this system has proven to be a difficult task. If the analysis of high order DPLL systems is to be achieved then the analysis technique needs to be simplified and not complicated. For this reason alone the nonlinear methods mentioned have proven to be unpopular with designers and linear methods are used exclusively. Second, any attempt at applying these nonlinear methods to the DPLL or APLL always start by assuming that the PD can be replaced

by a linear subtractor component. This is also the result of the APLL linearization of Section 3.2. Applying a nonlinear analysis technique to what is essentially a linearly approximated system is not very informative.

Behavioural models are a viable an accurate alternative to the traditional nonlinear and linear methods described thus far. They provide an efficient, workable and accurate nonlinear solution to the analysis of DPLL systems, these essentially model the nonlinear PFD component using a state transition diagram to track the changing states of the DPLL loop in a piecewise linear fashion. Behavioural methods as they are applied to the DPLL are considered in detail in the next section.

3.5 Behavioural Modelling of the DPLL

All of the DPLL design and modelling methods (linear and nonlinear) described up to now have been applied to the DPLL either in theory or in practice. However because of the complexity of the DPLL system it is more common in practice that simpler linear model analysis tools are utilised. This simplicity must be traded with the lack of accuracy of the linear model. For this reason the linear predictions of the system performance are always validated using a slow circuit level simulation as a final step in the design process. Behavioural models are found to be considerably more effective; modelling the CP-PFD nonlinearity without the unwieldiness of the Lyapunov derived nonlinear methods or the slowness of the circuit level solution, and are more accurate than the linear approximated model. Behavioural modelling techniques are discussed in detail in this subsection.

The complexity of the DPLL is due mainly to the fact that the CP-PFD is a time varying system exhibiting a quantization-like effect. The DPLL analysis is further complicated by the fact that the variable of interest around the loop changes from phase to voltage, which produces a large number of harmonics that are required to accurately model the digital nature of the waveforms [56]. Linear stability criteria such as [13, 26, 43] do not provide a definitive prediction of stability for the DPLL; in particular they ignore these nonlinearities which are crucial to the operation of the DPLL, however as mentioned earlier they can provide a starting point for the empirical design process illustrated in Figure 3.7 earlier.

The empirical nature of the DPLL design process means that more than one simulation will ultimately be required to determine the system performance. Replacing the circuit level simulations with behavioural models can significantly reduce the design time. Such behavioural models [3, 49, 50] consider the performance of the individual DPLL blocks, as opposed to every individual component in the case of a circuit level simulation. To explain this further consider the CP-PFD component block, behavioural models consider the expected behaviour of the CP-PFD output for all possible inputs, this can be extremely advantageous for systems that only have a small number of possible inputs. This is the case with the CP-PFD which has three possible input states and thus three possible corresponding outputs. These are outlined in the table below.

Input	**CP-PFD Output**	**CP-PFD State**
F_R Leads F_V	$+I_P$ Amps	UP
F_R Lags F_V	$-I_P$ Amps	DOWN
F_R and F_V Correspond	0 Amps	Null

Table 3.1 CP-PFD Input/Output States

Behavioural modelling techniques as applied to the DPLL work on the basis that the CP-PFD component operates in one of three possible states. The loop response is linear between these nonlinear state transitions. Thus the current output of the CP-PFD can be determined if the present CP-PFD state is known. The CP-PFD state changes depending on certain events occurring in the DPLL loop, it will depend on the occurrence of falling edges of the F_R and F_V digital signals. The next event is determined by monitoring the current phase of F_R and F_V, with the knowledge that the next falling edge event of the relevant signal will occur at a phase of 2π radians. If the falling edge R$\downarrow$ leads that of V$\downarrow$ then the PFD will move into the up state (U), if R$\downarrow$ lags V$\downarrow$ then the PFD state will be down (D), and finally if both signals correspond then the PFD state will be null (N). These state transitions were discussed earlier in Chapter 2 and are illustrated in the state transition diagram of Figure 2.9.

Using this state transition diagram the behavioural models use a set of difference equations to model the expected behaviour of the overall DPLL system, including the non-linear quantization like effect of the CP-PFD. Due to the fact that the DPLL system performance is linear between the nonlinear transitions of the CP-PFD, the system can be modelled using a set of linear difference equations describing the response of the system between each transition. Thus behavioural models change between three sets of difference equations depending on the present state of the PFD. These difference equations can be derived in a number of ways, two of which are presented in following subsections.

This behavioural modelling methodology of the PFD accurately models the response of the system without the need for a slow circuit level simulation. The state transition diagram is piecewise linear in nature and will be used in Chapter 5 to develop a novel design methodology for the DPLL. The significant advantage of behavioural models over linear theory is the accuracy of the model, while the advantage over a circuit level simulation is the speed of simulation. However the one drawback of the behavioural model is the complexity of the model equations. This is due to the existence of high order differential terms, thus it becomes mathematically infeasible to determine the system equations in closed form. This is found to be the case for loop orders of greater than four. In Chapter 4 a solution to this is proposed in the form of a charge approximation model. This model offers a simple approximation of the charge on the loop filter capacitors, which simplifies the difference equations, and removes the differential terms.

The behavioural model takes two significant forms, either a uniform or non-uniform sampling model, a seminal example of each of these methods Van Paemel [3] and Hedayat [50] are presented in the next two subsections and are then compared against a reference circuit level simulation to prove their accuracy.

3.5.1 Van Paemel Event Driven Model

The falling edge events of the VCO output signal described in the previous subsection occur at non-uniform time intervals due to the varying frequency of the VCO signal. In contrast to this characteristic, Van Paemel's model [3] is a discrete

time uniformly sampled model. The model samples independently of the DPLL events, while still using the state transition model of Figure 2.9. Uniform sampling simplifies the analysis of the second order model; however this increases the modelling time while reducing the accuracy relative to a non-uniform equivalent. The inaccuracy of the uniform approach is due to the fact that the time between events in the PLL loop is non-uniform. Therefore a variable sampling time is required which is much less than $1/F_R$.

The Van Paemel model considers the second order DPLL and therefore uses two state variables, τ the pulse width of the phase detector output pulse (which is directly related to the phase error), and V_C the voltage stored across the loop filter capacitor and resistor, as in Figure 2.18 earlier. The model equations describe the system state variables at the current time period from previous values of the state variables. This is carried out either from a set of difference equations, if they can be derived in closed form, or by numerical iteration. Van Paemel models the pulses of the PFD output – the behaviour of the system. These pulses are directly related to the value of V_C, so V_C can be estimated if the current state, next event and pulse width are all known. The shape of V_C is illustrated in Figure 3.20, an estimate of $V_C(k+1)$ can be determined using integration, or by determining the area under the V_C line in Figure 3.20.

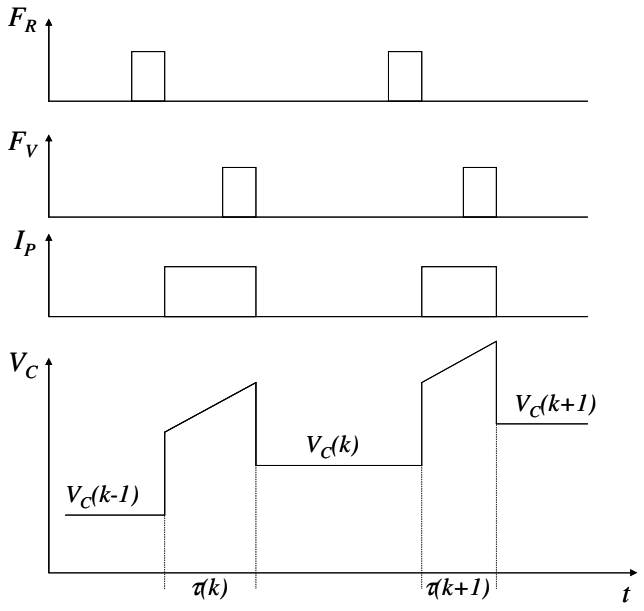

Figure 3.20 Plot of the CP-PFD Output Pulses and the Control Voltage Inputs

Van Paemel uses a prediction of the V_C pulse shape to determine the integral, there are four different possible pulse width shapes, these are the four cases outlined below.

1. Case 1 - $\tau(k)>0$ and $\tau(k+1)>0$

$$\tau(k+1)=\frac{-I_P R_1 - V_{C1}(k) + \sqrt{\left(I_P R_1 + V_{C1}(k)\right)^2 - \frac{2I_P}{C_1}\left(V_{C1}(k)(T-\tau(k)) - \frac{1}{K_V}\right)}}{\frac{I_P}{C_1}} \tag{3.34}$$

2. Case 2 - $\tau(k)<0$ and $\tau(k+1)<0$

$$\tau(k+1)=\frac{\frac{1}{K_V} - I_P R_1 \tau(k) - \frac{IP\tau^2(k)}{2C_1}}{V_{C1}(k)} - T - \tau(k) \tag{3.35}$$

3. Case 3 - $\tau(k)>0$ and $\tau(k+1)<0$

$$\tau(k+1)=\frac{1}{K_V V_{C1}(k)} - T + \tau(k) \tag{3.36}$$

4. Case 4 - $\tau(k)<0$ and $\tau(k+1)>0$

$$\tau(k+1)=\frac{-I_P R_1 - V_{C1}(k) + \sqrt{\left(I_P R_1 + V_{C1}(k)\right)^2 - \frac{2I_P}{C_1}\left(V_{C1}(k)\tau_{case2}(k+1)\right)}}{\frac{I_P}{C_1}} \tag{3.37}$$

For each of the above cases two difference equations are used to calculate the next set of values of $V_C(k+1)$ and $\tau(k+1)$. For example, for each iteration $V_C(k+1)$ can be calculated by determining which of these four cases has occurred and estimating $V_C(k+1)$ from the difference equation associated with that particular case. The final set of difference equations for $\tau(k+1)$ are given in equations (3.34-3.37). The second state variable $V_C(k+1)$ is determined from the calculated values of $\tau(k+1)$ as given in equation (3.38).

$$V_{C1}(k+1) = V_{C1}(k) + \frac{I_P}{C_1}\tau(k+1) \tag{3.38}$$

There are two additional cases possible that need to be considered, these help to simulate the out-of-lock behaviour of the loop. These two additional cases are outlined in [3].

Using the stability limit equations as defined in [27], Van Paemel calculates a linearised stability limit which places a restriction on the gain of the system. It is suggested that if this restriction is adhered to, then the stability of the system is guaranteed. This however is not the case. As we would expect from the observations of Section 3.2.2, the linear model cannot guarantee the stability of the nonlinear DPLL. Nevertheless the stability restriction as defined by Van Paemel is given in equation (3.39)

$$K_N = \frac{2}{1 + \frac{1}{2\tau_2 N}} \tag{3.39}$$

where K_N is the normalised loop gain and is equal to $I_P R_2 K_V T$. This boundary gives similar results to that of equation (3.12) earlier. This is to be expected as they are both derived from the same linear PLL model equations.

3.5.2 Hedayat et al. Event Driven Model

The second and third order event driven models of Hedayat et al [49, 50] also utilise the event driven behavioural modelling technique. However unlike the Van Paemel model, the time period *t* is a non-uniform sampling time and represents the time until the next falling edge event occurs.

The phase of the reference and VCO signals play an important role in the operation of this model. The loop is considered to be locked when the time difference between the reference and VCO signals is zero. This is the case when R↓ and V↓ occur sequentially or when the phase of both signals correspond.

If the phase of both these signals is known at some time t, then it is possible to determine the phase of these signals at time $t+\tau$ by integration. The falling edges R↓ and V↓ both occur when the reference is equal to 2π. Therefore it is possible to find the event times t_R and t_V when R↓ and V↓ occur respectively. Knowing the current DPLL state, the time of the next event and using the state transition diagram of Figure 2.9 it is possible to determine the value of V_C and therefore the VCO output signal from its characteristic equation (3.40), where F_{FR} is the VCO free running frequency and K_V is the VCO gain component.

$$F_V = K_V V_C + F_{FR} \tag{3.40}$$

The next PFD state and the control voltage V_C of the second order loop at time t_{n+} can be determined from Table 3.2, depending on the previous PFD state and the current event.

PFD @ tn-	**Event at t_n**	**PFD @ t_{n+}**	***VC(t_{n+})***
Null	R↓	Up	$V_C(t_{n-})+R_2*I_P$
Up	R↓	Up	$V_C(t_{n-})$
Up	V↓	Null	$V_C(t_{n-})-R_2*I_P$
Null	V↓	Down	$V_C(t_{n-})-R_2*I_P$
Down	V↓	Down	$V_C(t_{n-})$
Down	R↓	Null	$V_C(t_{n-})+R_2*I_P$

Table 3.2 Determination of VCO control voltage

The phase signals for this second order model are calculated as shown in equations (3.41) and (3.42).

$$\varphi_r(t^r_{n+1}) = \varphi_r(t_n) + f_r(t^r_{n+1} - t_n) \tag{3.41}$$

$$\varphi_V(t^v_{n+1}) = \varphi_V(t_n) + 2\pi \begin{bmatrix} \frac{\alpha K_V}{2}(t^{v\,2}_{n+1} - t_n^{\,2}) \\ +\left(F_{v0} - K_V\left(\alpha t_n - V_C(t_n^+)\right)\right)(t^v_{n+1} - t_n) \end{bmatrix} \tag{3.42}$$

In a similar manner to Van Paemel the determination of the next event in the case of a third or higher order model is not a straight-forward mathematical solution; a closed form solution to the calculation of t_v is not possible, so numerical iteration is used. This is achieved by numerical integration using the Newton-Cotes trapezium rule. The third order model uses a derived difference equation to define the value of $V_C(t_{n+})$. These equations are as shown in equations (3.43-3.45) in the order they need to be solved for each iteration.

$$t_{n+1}^{r} = t_n + \frac{2\pi - \varphi_r(t_n)}{F_R} \tag{3.43}$$

$$\varphi_V(t_{n+1}) = \varphi_V(t_n) + 2\pi \left[\begin{array}{l} \dfrac{K_V}{C_3}\left(\dfrac{i_2(t_n) - \beta_2}{\beta_1^2}(1 - e^{\beta_1(t_{n+1}-t_n)}) + \dfrac{\beta_3}{2}(t_{n+1}-t_n)^2 \right) \\ + \left(F_{V0} + K_V \left(\dfrac{i_2(tn) - \beta_2}{C_3\beta_1} + V_C(t_n) \right)(t_{n+1}-t_n) \right) \end{array} \right] \tag{3.44}$$

$$\varphi_r(t_{n+1}) = \varphi_r(t_n) + 2\pi F_R(t_{n+1} - t_n) \tag{3.45}$$

Hedayat et al. do not go as far as to develop the fourth order system equations. However using a similar technique to [49] and [50] it is possible to derive the fourth order equations using the filter structure of Figure 2.20. These equations, derived here and given in equations (3.46 – 3.48), replace the equivalent ones in [50] to increment the model to fourth order. To generate this model, first consider the current through each capacitor in the circuit, i_2, i_3, and i_4:

$$i_2(t_{n+1}) = \left(i_2(t_n) - \gamma_2\right)e^{-\gamma_1(t_{n+1}-t_n)} + i_2(t_n) + \gamma_6 \tag{3.46}$$

where, $\gamma_1 = \dfrac{C_2 + C_3 + C_4}{R_2C_2(C_3 + C_4)}$, $\gamma_2 = \dfrac{R_4C_4}{C_3 + C_4} i_p(t_n)$, and $\gamma_6 = \dfrac{R_4C_4}{C_3 + C_4} i_4(t_n)$

$$i_3(t_{n+1}) = \gamma_3 \left[\left(\gamma_2 - i_2(t_n)\right)e^{-\gamma_1(t_{n+1}-t_n)} + \gamma_4 + R_4C_4 \frac{di_4(t)}{dt} \right] \tag{3.47}$$

where, $\gamma_3 = \dfrac{C_3}{C_3 + C_4}$, and $\gamma_4 = i_p(t) - i_2(t) - \gamma_6$

$$i_4(t_{n+1}) = \gamma_5\left[\left(\gamma_2 - i_2(t_n)\right)e^{-\gamma_1(t_{n+1}-t_n)} + \gamma_4 + R_4C_3\frac{di_4(t)}{dt}\right] \tag{3.48}$$

where, $\gamma_5 = \dfrac{C_4}{C_3 + C_4}$.

Once each of the currents is known, the voltage across each capacitor is calculated as in equations (3.49-3.51). It can be seen from Figure 2.20 that the control voltage V_C is equivalent to the voltage across capacitor C_4.

$$Vc_2(t_{n+1}) = Vc_2(t_n) + \frac{1}{C_2}\left[\begin{array}{l}\dfrac{i_2(t_n)-\gamma_2}{\gamma_1}\left(1-e^{-\gamma_1(t_{n+1}-t_n)}\right)\\ +\left(i_2(t_n)+\gamma_6\right)\left(t_{n+1}-t_n\right)\end{array}\right] \tag{3.49}$$

$$Vc_3(t_{n+1}) = Vc_3(t_n) + \frac{\gamma_3}{C_3}\left[\begin{array}{l}\dfrac{\gamma_2-i_2(t_n)}{\gamma_1}\left(1-e^{-\gamma_1(t_{n+1}-t_n)}\right)\\ +\gamma_4\left(t_{n+1}-t_n\right)+R_4C_4i_4(t)\end{array}\right] \tag{3.50}$$

$$Vc_4(t_{n+1}) = Vc_4(t_n) + \frac{\gamma_5}{C_4}\left[\begin{array}{l}\dfrac{\gamma_2-i_2(t_n)}{\gamma_1}\left(1-e^{-\gamma_1(t_{n+1}-t_n)}\right)\\ +\gamma_4\left(t_{n+1}-t_n\right)+R_4C_3i_4(t)\end{array}\right] \tag{3.51}$$

Finally the phase error $\phi_e(t_{n+1})$ is calculated from $\phi_R(t_{n+1})$ - $\phi_V(t_{n+1})$, where $\phi_R(t_{n+1})$ is the phase of F_R and $\phi_V(t_{n+1})$ is the phase of signal F_V and is calculated as in equation (3.52).

$$\begin{aligned}\varphi_v(t_{n+1}) = \varphi_v(t_n) + \\ +2\pi\left\{\begin{array}{l}\dfrac{K_V\gamma_3}{C_3}\left[\dfrac{i_2(t_n)-\gamma_2}{\gamma_1^2}\left(1-e^{-\gamma_1(t_{n+1}-t_n)}\right)+\dfrac{\gamma_4}{2}\left(t_{n+1}-t_n\right)^2\right]\\ +\left[fvo + K_V\left(\dfrac{\gamma_3}{C_3}\left(\dfrac{\gamma_2-i_2(t_n)}{\gamma_1}+R_4C_4i_4(t)\right)+Vc(t_n)\right)\right]\\ \times\left(t_{n+1}-t_n\right)\end{array}\right\}\end{aligned} \tag{3.52}$$

Now that we can accurately model fourth order it is possible to revisit the rule-of-thumb for fourth order systems that was suggested in Section 2.4.1 earlier. This

stated that the choice of C_4 should be no greater than 1/10th of C_3. This can be validated by plotting the transient response of the fourth order model as the capacitor C_4 is varied. Figure 3.21 illustrates that as the value of capacitor C_4 approaches C_3 the system lock time is steadily reduced, however a threshold is reached at C_3/C_4 is approximately equal to 10. Beyond this threshold the lock time increases dramatically causing the system to become unstable. This is due to the additional fourth order pole component encroaching on the third order system poles. This is to be expected because with the fourth order pole placed out-of-band, it has minimal effect on the overall transients and therefore stability of the DPLL system. However as the additional fourth order pole encroaches on the other system poles, it effects the system transients and therefore needs to be considered in terms of the loop stability.

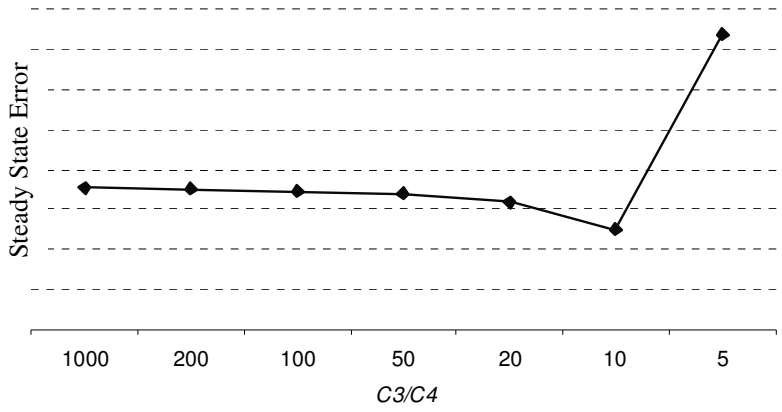

Figure 3.21 Plot of Steady State Error as C_4 Approaches C_3

Thus the rule-of-thumb that C_4 should be chosen no greater than 1/10th the value of C_3, proves reasonable in light of the above result. If this rule-of-thumb is ignored and the value of C_4 is chosen too close to the value of C_3, then the traditional third order linear stability criterion cannot be safely applied to this fourth order system.

Finally it is worth pointing out that even though the Hedayat model has been extended to fourth order here; it is mathematically infeasible to increment the model to orders higher than fourth. The model equations cannot be found in closed form due to the existence of high order differential terms.

3.6 Summary

In this chapter the simulation, modelling and analysis of the DPLL was considered in detail. It was discussed that the DPLL system performance can be determined by using a number of methods; these are a linear approximation, circuit level simulation, nonlinear prediction or using a behavioural model.

The linear approximation of the DPLL was derived in Section 3.2; it was shown that this approximation is actually an approximation of the APLL with a mixer PD. The application of this same approximation to the DPLL can be justified if the CP-PFD behaviour is ignored and the state variables (control voltage and phase error) are considered equivalent in both systems. The limitations of the linear approximation are that it is only valid for a small phase error when the DPLL is close to lock, and when the loop filter bandwidth is chosen to be less then one tenth of the reference signal frequency. This means that there will always be some error in the linear approximation of the DPLL due to the fact that the nonlinear CP-PFD behaviour is not considered. The significant advantage of the linear method is due to the analytical nature of the analysis and the advanced and powerful nature of linear analysis tools – this alone explains its continued popularity.

In Section 3.3 circuit level simulations were discussed, it was concluded that these methods, while accurate, are extremely slow and can only be used empirically to determine the desired DPLL performance. Thus circuit level simulators are generally used only as a final step in the design process to validate the system performance.

Alternative methods to the linear approximation are nonlinear techniques which are found to unnecessarily complicate the analysis of the DPLL. It was discussed that the application of such nonlinear methods to the DPLL were rare in the literature for this reason; instead such methods were applied only to the APLL, which were subsequently linearised to simplify the analysis. Despite the less complex nature of the APLL relative to the DPLL, these analyses were still complex.

The final DPLL analysis method discussed in this chapter was behavioural or event driven modelling methods. Such methods offer advantages over other methods in that they accurately model the behaviour of the DPLL system – including the behaviour of the nonlinear CP-PFD component block, they are significantly faster than the circuit level simulation, and they model the DPLL behaviour on a component block level as opposed to transistor level. Finally they do not overcomplicate the system equations as in the case of existing nonlinear methods, allowing insight into their behaviour to be gained.

Two existing behavioural models were considered in detail, the first is proposed by Van Paemel. This model graphically determines the control voltage of a second order DPLL from the integral of the output response of the loop filter; the second state variable is the uniform sampling time τ. This technique has two drawbacks:

1. It estimates the time of the next event. However as the frequency of the VCO output signal varies between these events this estimation becomes less accurate. To improve the accuracy of the model the time of the next event needs to be recalculated at each sample point.
2. Van Paemel considers only the second order case as this is a simple system using a first order filter; this filter response can be solved using first order linear equations. Higher order system equations become impossible to solve in closed form due to the presence of high order differential terms.

The second behavioural model considered is Hedayat; this model improves on the first drawback of Van Paemel by using a non-uniform sampling time, which corresponds to the time until the next event, increasing the model accuracy significantly. In the case of high order DPLL modelling the same drawback as outlined for Van Paemel also exists with this method.

Thus in conclusion behavioural modelling techniques offer the fastest and most accurate method of analysing the DPLL. However the disadvantage of existing behavioural models is that they become increasingly complex as the loop order is increased. In fact it is found that for orders greater than four, the system model equations cannot be determined in closed form. A solution to this issue is proposed in the next chapter.

CHAPTER 4
A NON-LINEAR ANALYSIS TECHNIQUE FOR MODELLING HIGH ORDER DIGITAL PHASE LOCKED LOOPS

4.1 Introduction

Mixed signal systems such as the DPLL, by definition, contain both analogue and digital elements. Trying to model mixed signal systems analytically presents a significant challenge due to the differences between digital and analogue signal representations [57], i.e. the digital part of the circuit uses discrete variables, while the analogue part uses continuous variables. Thus there is a requirement to convert the discrete signal into a continuous signal, within the loop, and then back to digital [3, 49, 50, 58, 59]. This is due to the transition between states as discrete events occur within the loop, this signal transformation means that these mixed signal devices exhibit quantization-like effects which are inherently nonlinear. This, in essence, means that these systems cannot be accurately analysed using traditional linear methods because they do not consider the quantization effects of the DPLL.

In recent years behavioural models, as outlined in the previous chapter, have become increasingly popular because of the inaccuracies and impracticalities associated with the linear and circuit level simulation methods. These behavioural methods can accurately model the nonlinearities in a significantly shorter computation time. They work by modelling, at a high level, the behaviour of each individual component block within the system. Such mixed signal behavioural analysis methods work well for low orders, the difficulty occurs when higher orders are considered. It is found that for high orders, it is mathematically infeasible to determine closed form model equations due to the existence of high order differential terms.

The differential terms, which are due to the parasitic elements in the loop, need to be solved in order to determine the systems difference or behavioural equations either in

closed form or in to some numerically simple form. In this chapter a novel approach to the derivation of the system equations is presented, this method approximates the loop filter capacitor charge which results in system equations that do not contain differential terms, thus making it possible to derive the behavioural equations in closed form. Without such an approximation it would not be possible to determine the high order DPLL behavioural equations. This charge approximation methodology is presented in the next section, using a simple first and second order RC filter as an example. While the application of the proposed charge approximation to such RC filters is not particularly beneficial (due to the fact that the filter is analogue and its behavioural equations can easily be found in closed from without the need of any approximation) the RC filter is used purely as an example to explain the basic principles of the charge approximation. Once some understanding of the charge approximation methodology is gained, it can then be applied to more complex mixed signal systems such as the DPLL, this is carried out in Section 4.3, where the behavioural equations are determined for orders of two up to five

It is worth pointing out that the charge approximation methodology that is presented in this chapter is applicable to all mixed signal systems. Along with the DPLL another example of mixed signal systems is the sigma delta modulator (SDM). The SDM is used in devices such as analogue to digital convertors (ADC) and digital to analogue convertors (DAC). Both the SDM and the DPLL are feedback loops containing both discrete time and continuous time signals. The major difference between the two systems is that the SDM has a fixed discrete signal sample rate, whilst the DPLL has a variable sample rate. This means that the analysis of the DPLL is slightly more complex then that of the SDM. This chapter only considers the application of this methodology to the DPLL system. The application of the charge approximation to other mixed signal systems is beyond the scope of this thesis.

The introduction of such a simplifying approximation inevitably results in the introduction of an additional amount of error. It is expected however, due to the fact that the error is bounded by the reference frequency, that this error will be small relative to the continuous time approximation of the linear model, and that it will be minimised as the time between samples decreases. This error and the limitations that

it imposes on the model are considered in detail in Section 4.4. The charge approximation model is also compared to existing behavioural methods and a circuit level simulation to show that any introduced error can be minimised.

4.2 The Charge Approximation Methodology

As discussed the existence of high order differential terms in high order DPLL system equations means that it is not mathematically feasible to determine these equations in closed form. The numerical resolution of such high order polynomials is difficult. To help explain the proposed approach we initially take as an example a simple first order RC transconductance filter shown in Figure 4.1 below.

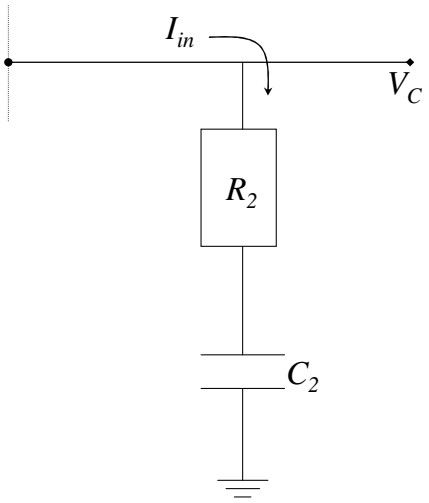

Figure 4.1 First Order RC Transconductance Filter Structure

This filter structure is similar to the traditional passive loop filter structure of the second order DPLL with a charge pump PFD component. It is transconductance in nature because there is a requirement within the DPLL loop to convert the current output of the charge pump PFD to a voltage, for operation by the VCO. The system equation for this filter is calculated using equation (4.1) below.

Voltage across Resistor Voltage across Capacitor

$$V_C(t_{k+1}) = R_2 I_{in} + \frac{1}{C_2}\int I_{in}\,dt \qquad (4.1)$$

As mentioned earlier the input current to the filter, I_{in}, varies in a discrete-like manner as the system state changes (Table 3.1). Thus for any one period of time, Δt, the input will be a constant value, either $+I_P$, $-I_P$, or 0 amps. Thus if I_{in} is a constant current source, and assuming zero initial conditions, then equation (4.1) can be solved by using the trapezium rule of Newton-Cotes numerical integration formulas and thus be rewritten as in equation (4.2), without the integral term:

$$V_C(t_{k+1}) = R_2 I_{in} + \frac{\Delta t I_{in}}{C_2} \tag{4.2}$$

where Δt is the length of time that the current is pumped into the filter. In this example the solution is straight forward, as there are no differential terms. The control voltage can be solved in closed form for a given I_{in}, which varies depending on the DPLL loop state.

In the case of a second order loop filter, the order is increased by including an additional capacitor as in Figure 4.2. Again this filter is a passive transconductance filter structure similar to the loop filter of a third order DPLL.

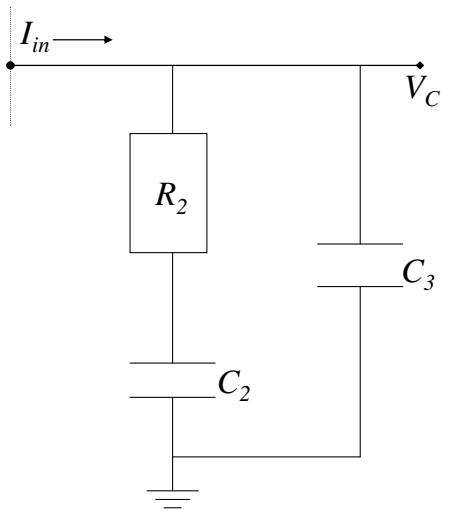

Figure 4.2 Second Order Filter

The system equation for this second order filter contains both an integral and a differential term as shown in equation (4.3) below.

$$V_C(t_{k+1}) = \frac{1}{C_2}\int I_{in}dt + I_{in}R_2 - R_2C_3\frac{dV_C}{dt} - \frac{C_3}{C_2}V_C(t_{k+1}) \tag{4.3}$$

Again I_{in} is constant within any time period Δt, thus the integral of I_{in} can be solved as in the first order case earlier (i.e. it is equal to $\Delta t I_{in}$). In this case it is not quite as straight forward to solve equation (4.3) in closed form, due to the presence of the differential term which needs to be solved. As the filter orders are increased so do the differential orders, these high order differentials cannot be solved into closed form, this will become evident later when the fourth order DPLL is considered.

However in the case of equation (4.3) the first order differential can be resolved into closed form, this solution is derived as follows.

The differential term in equation (4.3) can be calculated from the knowledge of the slope of the output voltage V_C, i.e. $\frac{dV_C}{dt} = \frac{V_C(t_{k+1}) - V_C(t_k)}{\Delta t}$. Thus equation (4.3) can be rewritten as in equation (4.4)

$$V_C(t_{k+1}) = \frac{I_{in}\Delta t}{C_2} + I_{in}R_2 - \frac{R_2C_3}{\Delta t}\left(V_C(t_{k+1}) - V_C(t_k)\right) - \frac{C_3}{C_2}V_C(t_{k+1}) \tag{4.4}$$

Moving all the $V_C(t_{k+1})$ terms to the left hand side and solving, the control voltage behavioural equations can be found in closed from as given in equation (4.5).

$$V_C(t_{k+1}) = \frac{I_{in}\Delta t + I_{in}R_2C_2 - \frac{R_2C_2C_3}{\Delta t}V_C(t_k)}{C_2 + C_3 + \frac{R_2C_2C_3}{\Delta t}} \tag{4.5}$$

The above derivation of equation (4.5) is carried out in a similar manner to that of the two behavioural models described in Chapter 3 [3, 49], i.e. replace the first order differential term with the slope of the control voltage and solve the resulting expression for $V_C(t_{k+1})$.

As mentioned earlier, an increase in the order of the system means a growth in the number of differential terms, for example expect that a third order transconductance filter will have a number of differential terms with orders up two. In other words the

third order equivalent of equation (4.3) will have a d^2V_C/dt^2 term that will need to be solved. Likewise for a fourth order transconductance filter we expect to find a third order differential term. To numerically solve these equations an approximation is required for high orders which removes or in some way approximates these differential terms, thus making it possible to increment the model to high orders. The solution proposed here is that these high order differential terms can be removed using a charge approximation, thus simplifying the behavioural equation analysis. This charge approximation technique simplifies the derivation of the system behavioural equations by approximating and removing the differential terms. This is achieved by initially calculating the voltage across the capacitor C_2 using the charge expression of equation (4.6), rather than the integral expression of the voltage on the capacitor as in equation (4.7)

$$V_{C_2}(t_{k+1}) = \frac{Q_{C_2}(t_k) + \Delta Q}{C_2} \tag{4.6}$$

$$V_{C_2}(t_{k+1}) = \frac{1}{C_2}\int_{t_k}^{t_{k+1}} I_2 dt \tag{4.7}$$

To solve equation (4.6) a determination of the change in the loop filter capacitor charge, ΔQ, over a given time period Δt is required. But $\Delta Q = Q_{C2}(t_{k+1})-Q_{C2}(t_k)$, thus to determine $V_{C2}(t_{k+1})$ we need to know $Q_{C2}(t_{k+1})$, which we do not know at this stage. However $V_{C2}(t_{k+1})$ can be approximated by making the following approximation of ΔQ:

$$\Delta Q = \overline{I_1}\Delta t \approx I_1(t_k)\Delta t \approx I_1(t_{k+1})\Delta t \tag{4.8}$$

where $\overline{I_1}$ is the mean current through C_1 during the time period Δt. This charge approximation is illustrated in Figure 4.3 below.

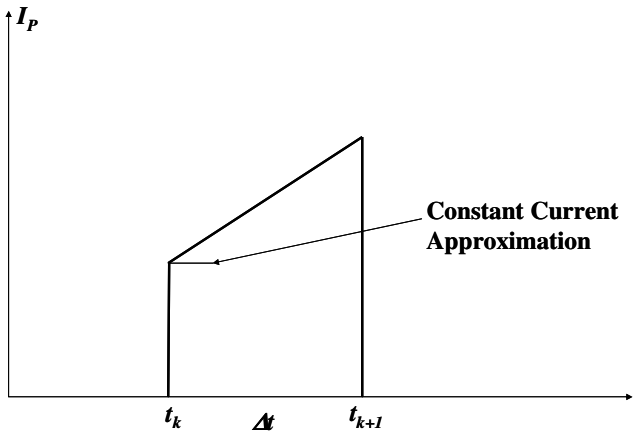

Figure 4.3 Current Approximation that Leads to the Charge Approximation

Thus equation (4.6) given above can now be approximated by equation (4.9).

$$V_{C_1}(t_{k+1}) = \frac{Q_{C_1}(t_k) + I_1(t_k)\Delta t}{C_1} \tag{4.9}$$

Using this approach, the system's behavioural equations no longer contain differential or integral terms, and only contain terms that are known at time t_k. Consider again the first order filter with the structure of Figure 4.1 and the system equation as given in equation (4.1). This filter may be charge approximated as follows: first, equation (4.1) is rewritten with the charge on the capacitor expressions, as in equation (4.10).

$$V_C(t_{k+1}) = RI_{in} + \frac{Q(t_k) + \Delta Q}{C} \tag{4.10}$$

Assuming we start with an initial charge $Q(t_k)$ equal to zero, then the voltage at the next time period $V_C(t_{k+1})$ can be rewritten as:

$$V_C(t_{k+1}) = RI_{in} + \frac{\Delta Q}{C} \tag{4.11}$$

So to determine $V_C(t_{k+1})$ it is necessary to calculate the change of capacitor charge ΔQ during the time period Δt. For a first order filter this is a simple linear calculation, as the average current, $\overline{I_{in}}$, flowing into the capacitor during the time

period Δt is constant. Thus the charge approximated output voltage of the filter is as given in equation (4.12) given below.

$$V_C(t_{k+1}) = R_2 I_{in} + \frac{I_{in}\Delta t}{C} \tag{4.12}$$

Both the integral filter equation, approximated earlier in equation (4.2) using Newton-Cotes and the charge approximated filter of equation (4.12) are the same. This is to be expected in this case for two reasons:

1. Because we are using a first order filter, the Newton-Cotes solution to the integral in equation (4.2) is an accurate linear estimate of the integral.
2. In terms of the charge approximation made in equation (4.12), the input current I_{in} is constant, therefore the average current through the capacitor is in fact equal to the initial current.

4.2.1 Behavioural Equation Derivation using the Charge Approximation

The effects of including a charge approximation and derivation of the system behavioural equations can be better illustrated by considering again the second order filter of Figure 4.2 with the system equation as given in equation (4.3). In this case the second order filter equation includes both an integral and a differential term. To derive the behavioural equations using the charge approximation carry out the following steps:

1. Assign values to all the input system nodes (V_A, V_B, V_D[v] etc) in the filter as in Figure 4.4, and determine the voltages across each section in terms of the capacitor charge as in equation (4.13 – 4.14).

---

[v] Note that the voltage expression V_C has not been used here intentionally because V_C is generally the voltage at the filter output node.

$$V_A(t_{k+1}) = R_2 I_2(t_{k+1}) + \frac{Q_2(t_k) + \overline{I}_2 \Delta t}{C_2} \tag{4.13}$$

$$V_B(t_{k+1}) = V_C(t_{k+1}) = \frac{Q_3(t_k) + \overline{I}_3 \Delta t}{C_3} \tag{4.14}$$

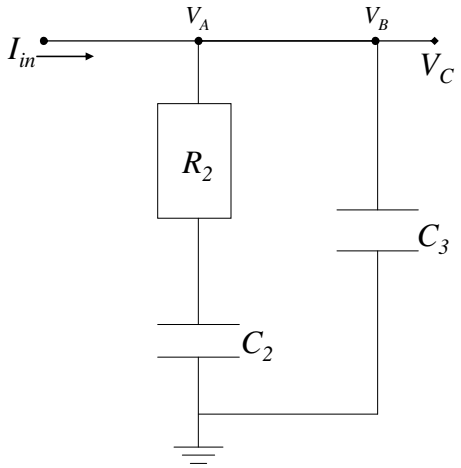

Figure 4.4 Assignment of Nodes

2. Equate each of the node voltages as in equation (4.15) below.

$$I_2(t_{k+1})R_2 + \frac{Q_2(t_k) + \overline{I}_2 \Delta t}{C_2} = \frac{Q_3(t_k) + \overline{I}_3 \Delta t}{C_3} \tag{4.15}$$

Note that in this case there is just one resulting equation (i.e. $V_A = V_B$). In the case of higher order filters there will be more expressions, these will come from the extra voltage nodes, i.e. in the case of the fourth order by equating nodes V_A=V_B V_A=V_D and V_D=V_B.

3. Including the input current expression given in equation (4.16), and approximating the average currents in equation (4.15) to $I(t_{k+1})$, (the current at time t_{k+1}), we are left with two simultaneous equations (equations (4.15 – 4.16)) and two unknowns ($I_2(t_{k+1})$ and $I_3(t_{k+1})$).

$$I_{in} = I_2(t_{k+1}) + I_3(t_{k+1}) \tag{4.16}$$

4. Solve these simultaneous equations for each current expression as follows:

Let this be equal to ρ

$$I_2(t_{k+1})\left(R_2+\frac{\Delta t}{C_2}\right)=\frac{I_{in}\Delta t}{C_3}-\frac{I_2(t_{k+1})\Delta t}{C_3}-\overline{\left(\frac{Q_2(t_k)}{C_2}-\frac{Q_3(t_k)}{C_3}\right)} \tag{4.17}$$

$$I_2(t_{k+1})\left(R_2+\frac{\Delta t}{C_2}+\frac{\Delta t}{C_3}\right)=\frac{I_{in}\Delta t}{C_3}-\rho \tag{4.18}$$

$$I_2(t_{k+1})=\frac{I_{in}\Delta t C_2-C_2C_3\rho}{R_2C_2C_3+\Delta t\left(C_2+C_3\right)} \tag{4.19}$$

5. Finally now that we know $I_2(t_{k+1})$, it is possible to calculate $I_3(t_{k+1})$, the charge on each filter capacitor, and the filter output voltage using equations (4.20 – 4.23) as given below.

$$I_3(t_{k+1})=I_{in}-I_2(t_{k+1}) \tag{4.20}$$

$$Q_2(t_{k+1})=Q_2(t_k)+\Delta t I_2(t_{k+1}) \tag{4.21}$$

$$Q_3(t_{k+1})=Q_3(t_k)+\Delta t I_3(t_{k+1}) \tag{4.22}$$

$$V_C(t_{k+1})=\frac{Q_3(t_{k+1})}{C_3} \tag{4.23}$$

Thus equations (4.19 – 4.23) are the charge approximated behavioural equations for the second order filter. Though this solution is more expansive than the solution of equation (4.5) earlier, it has no differential terms. The system behavioural equation derivation can also be applied, as it stands, to all orders of transconductance filters and DPLL systems, i.e. by using the five steps outlined above.

The charge approximation has been used so far to determine the behavioural equations for first and second order transconductance filters, and it has been shown that these behavioural equations can be derived without the use of the charge approximation. The second order filter was derived here for two reasons: first, as a simple example to introduce the charge approximation; and second, as a reference to

compare the accuracy of the charge approximated equations. This comparison will be used in the next subsection to investigate the approximation error.

4.2.2 Charge Approximation Error of a Second Order Filter

In the previous section it was seen that by deriving the filter behavioural equations using the charge on the loop filter capacitors rather than the voltage at the nodes, a simplification can be made that approximates the change in charge ΔQ by assuming that the average current through a capacitor during time period Δt is approximately equal to the current at time t_{k+1}, see equation (4.24).

$$\overline{I}_1 \Delta t \approx I_1(t_k)\Delta t \approx I_1(t_{k+1})\Delta t \tag{4.24}$$

The advantage of making this approximation is that differential terms are removed from the system model equations thus the model behavioural equations can be determined in closed form. This approximation is at the expense of some additional modelling error which we have already mentioned is directly proportional to the time interval Δt. This error is now considered in detail.

First consider the charge approximation on the current flowing into one capacitor C_1 using equation (4.24), the error that is introduced is equal to the difference between the average current into capacitor C_1 and the current at time t_k, see Figure 4.5.

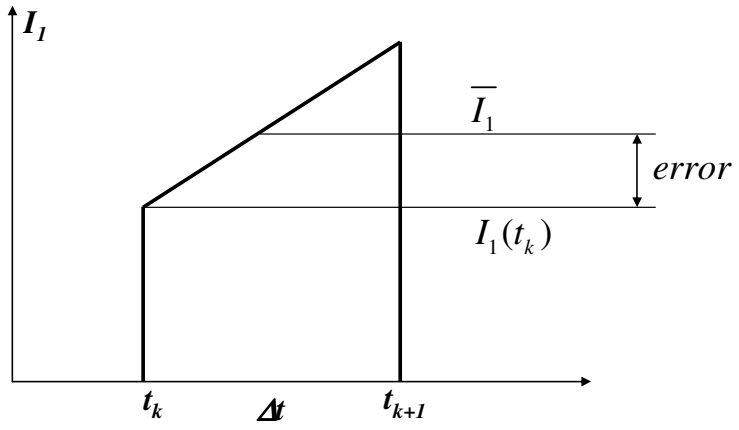

Figure 4.5 Estimate of the Charge Approximation Error

The approximated control voltage in terms of the additional approximation error can be calculated as follows:

$$V_C = \frac{Q_C + \Delta Q}{C_2} = \frac{Q_C + \bar{I}\,\Delta t}{C_2} \tag{4.25}$$

where V_C is the actual control voltage. Utilising the charge approximation in equation (4.25) means that the average current, $\bar{I}$, is replaced with the current at the end of the time period $I(t_{k+1})$. This results in:

$$V_{C\_Approx} = \frac{Q_C + I\left(t_{k+1}\right)\Delta t}{C_2} \tag{4.26}$$

Since $I(t_{k+1})$ is equal to $\bar{I}$ + I_{error}, where I_{error} is the additional error due to the approximation, then:

$$V_{C\_Approx} = \frac{Q_C + \left(\bar{I} + I_{error}\right)\Delta t}{C_2} = \frac{Q_C + \bar{I}\,\Delta t}{C_2} + \frac{I_{error}\Delta t}{C_2} \tag{4.27}$$

$$\Rightarrow V_{C\_Approx} = V_C + \frac{I_{error}\Delta t}{C_2} \tag{4.28}$$

Therefore the approximated control voltage V_{C_Approx} is equal to the actual control voltage V_C plus some small amount. This results in the approximated control voltage always being something greater than the actual control voltage. This fact will be used later in section 5.3.5 to show that the prediction of the stability boundary is more conservative than that of the actual boundary. This guarantees that any set of parameter values chosen within the piecewise linear boundary will be stable; this is in contrast to the linear prediction.

It can also be seen from equation (4.28) that the introduced error not only depends on the additional error current due to the approximation, I_{error}, but also depends on the time period Δt. So for low error Δt needs to be small compared to the loop filter time constant. The last term given in equation (4.28) is the approximation error due to one capacitor. More complex systems will have an approximation error that depends on the number of capacitors. Consider again the second order filter from earlier. In

Figure 4.6 the output voltage of a second order transconductance filter is plotted for a step in the input current. The filter output is determined using both the filter system equations (equation (4.5)) and the charge approximated equations (equations (4.19) up to (4.23)), as in Figure 4.6, it can be seen that the approximated response is less accurate as time step increases.

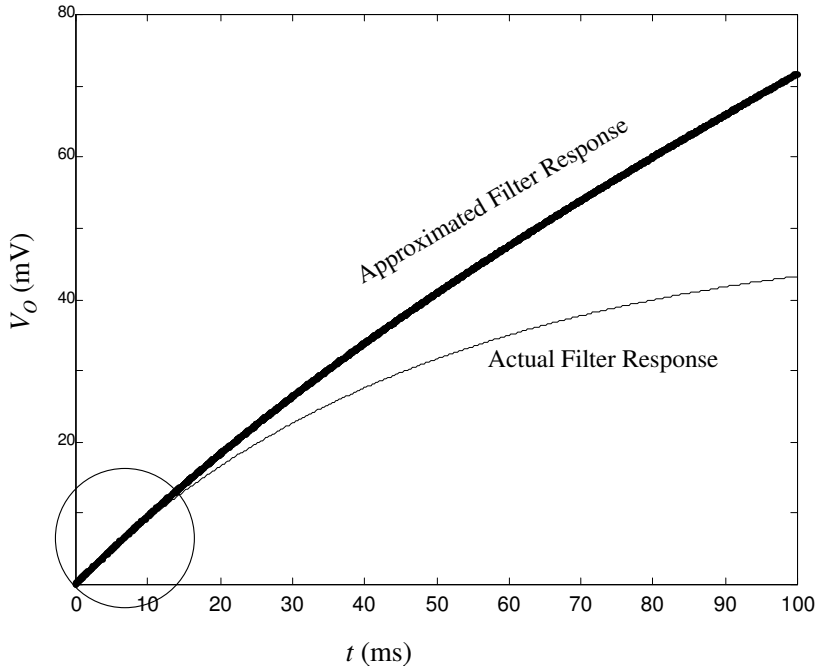

Figure 4.6 Second Order Filter Response

As is to be expected, Figure 4.6 shows that the difference between the voltage outputs of the actual and the approximated capacitor, to a constant current input, increases with time (Δt). This difference is the approximation error and is plotted in Figure 4.7, for three different capacitor input voltages and for a range of step sizes Δt. It can be seen that the error introduced due to the charge approximation on the RC filter reduces to zero as the time interval Δt is reduced.

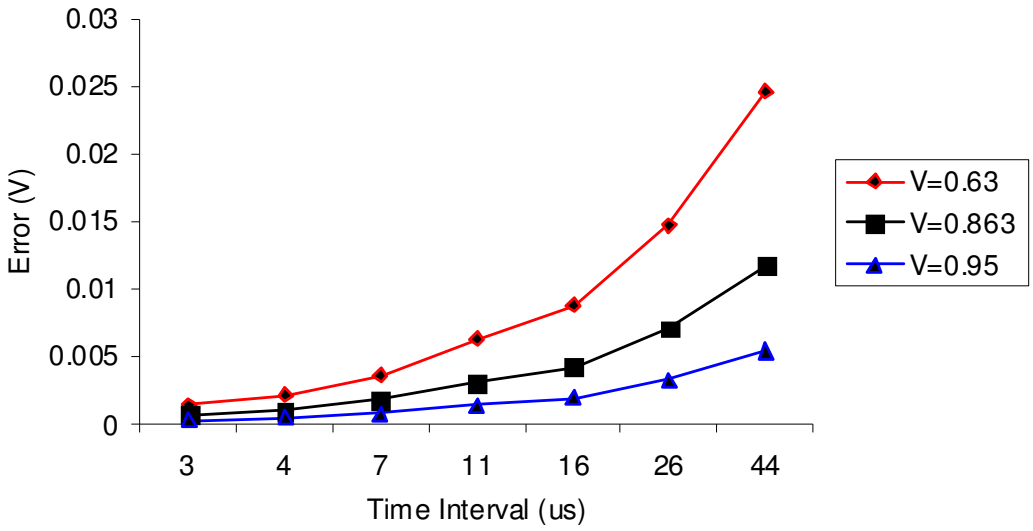

Figure 4.7 Increasing Error with Increasing Step Size at Three different Voltages

Thus the error is found to be bounded by the time interval Δt, as illustrated in Figure 4.8 below. Therefore the charge approximation methodology is only beneficial to systems that have a short Δt (Note that in this case Δt is not specifically the time period of the sampled system but the maximum length of time that the system pumps current into the capacitor).

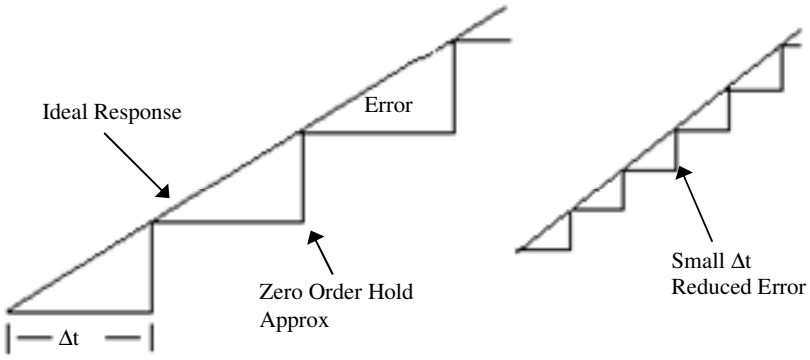

Figure 4.8 Zero Order Hold Approximation (a) with Large Δt, (b) Smaller Δt Smaller Error

It has already been discussed that the charge approximation provides little benefit to filter analysis; the major benefit of this approach is when it is applied to more complex mixed signal devices. One specific example of a system that will benefit from this approximation is the DPLL; this is because the DPLL, with a charge pump

PFD, pumps a constant current into the loop filter for only a fraction of the sample time. Thus the expected error due to the charge approximation is small. The charge approximation is applied to this DPLL system in the next section (orders of four and five) with a resulting reduction in the complexity of the system equations and a closed form expression for the behavioural equations.

4.3 Application of the Charge Approximation to High Order Digital Phase Locked Loops

As discussed previously existing mixed signal behavioural models [3, 49] are restricted mathematically to low orders. As the order is increased it becomes mathematically infeasible to determine the system equations in closed form due to the existence of high order differential terms. In this section a practical application of the charge approximation technique is applied to various DPLL systems. It will be shown that the charge approximation approach will reduce the complexity of the fourth and fifth order DPLL model equations and from this the system behavioural equations can be found in closed form. It is mathematically infeasible to determine the analogous behavioural model equations [3, 49, 58, 59] in closed form. This solution will be at an added cost of some bounded error which will be considered in Section 4.3.5.

4.3.1 Second Order DPLL Charge Approximation

Behavioural models, to summarise Section 3.5, model the mixed signal nature of a system by employing a state transition diagram to monitor the changing states of the system. This is due to the occurrence of discrete events within the loop. Take as an example the second order DPLL system with the structure as in Figure 4.9 below, with a first order loop filter similar to the filter of Figure 4.1, and a feedback divide ratio (÷N) of one.

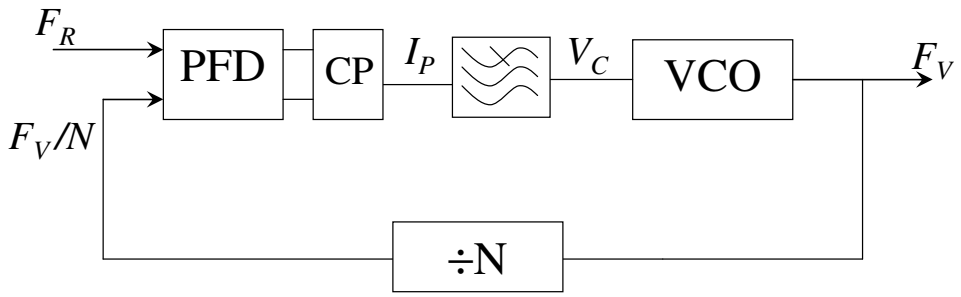

Figure 4.9 DPLL Loop Structure

Such a feedback system uses the state transition diagram of Figure 2.9, to track the state transitions of the PFD component, and thus determine the CP output current I_P. The PFD block starts in a predefined state (normally the Null state), the model tracks the state transitions of the loop as events occur. The events in question are the falling edges of the reference and VCO digital signals. To track these events some knowledge of the phase of the reference and VCO signals is required. The phase of each signal will cycle from zero up to 2π radians, when the phase of the relevant signal reaches 2π then a falling edge event of the respective signal has occurred. The phase of each signal can be monitored by using the following phase equations:

$$\phi_R(t_{k+1}) = \phi_R(t_k) + 2\pi F_R T \tag{4.29}$$

$$\phi_V(t_{k+1}) = \phi_V(t_k) + \frac{2\pi}{N}\left(K_V \int V_C dt + F_{FR} T\right) \tag{4.30}$$

where ϕ_R and ϕ_V are the phase of the reference and VCO signals respectively, N is the divide ratio and T is the sampling period. The integral of V_C in equation (4.30) is calculated using first order numerical integration as in equation (4.31).

$$\int V_C dt \approx T V_C(t_k) + 2\frac{V_C(t_{k+1}) - V_C(t_k)}{T} \tag{4.31}$$

The behavioural equation that is required to model the system depends on the present state of the loop. Within this state (i.e. between the occurrences of falling edge events) the mixed signal system is analogue only and thus a closed form analogue behavioural equation is used to model the system response. The occurrence of

events (this being the digital part) results in a change in the system state and thus a change in the analogue behavioural equation that is required to determine the systems response. Thus it is necessary to use a set of unique behavioural equations to model the system, one for each possible state. These behavioural equations contain differential terms which increase in order as the order of the loop filter increases. For high order systems there is no closed form solution for the behavioural equations and thus the behavioural model equations cannot be determined. Again taking the DPLL system model as an example, such a system has three states and a state transition diagram as in Figure 2.9. It will require three behavioural equations to model the system response. In the case of the DPLL these behavioural equations will only vary depending on the value of the PFD output current, i.e. $+I_P$, $-I_P$, or 0.

Equation (4.30) above determines the phase of the VCO output signal, this is dependant on the loop filter output. Thus the overall second order DPLL system can be modelled using the two phase equations (4.29 – 4.30) and the first order loop filter behavioural equation, which has already been calculated earlier in equation (4.12). The charge approximated set of equations are as given below:

$$V_C(t_{k+1}) = I_{in}R_2 + \frac{Q_2(t_{k+1})}{C_2} \tag{4.32}$$

$$Q_2(t+1) = Q_2(t) + I_{in}T \tag{4.33}$$

where $V_C(t+1)$ and I_{in} are the VCO control voltage and the loop filter input current respectively, as shown in Figure 4.9 earlier. The loop filter input I_{in} depends on the charge pump current output thus it is one of $+I_P$, $-I_P$, or 0 amps.

4.3.2 Third Order DPLL Charge Approximation

The third order DPLL contains a second order loop filter. Thus the third order DPLL charge approximation can be implemented using the second order filter order equations derived earlier. The control voltage V_C is determined using the charge approximated second order filter equations, rewritten in equations (4.34)-(4.36) below. The same state machine model and phase equations as in equations (4.29)

and (4.30) may be used. Note that using equation (4.31) to determine the integral of V_C for the third order system is not completely accurate because unlike the second order DPLL case the integral of V_C is not a straight linear solution. This can be overcome by using a piecewise linear solution, which is achieved by reducing the sample time to some integer fraction of T, i.e. the new sample time T' is equal to T/η where η is some integer. From this point forward it will be assumed that $T=T'$, which means that the model sample period used below is no longer the reference signal period, but some predefined sample time that is equal to the reference period divided by η.

$$Q_2(t_{k+1}) = Q_2(t_k) + \frac{T^2 C_2 I_P - TC_2C_3\left(\frac{Q_2(t_k)}{C_2} - \frac{Q_3(t_k)}{C_3}\right)}{T(C_2+C_3) + R_2C_2C_3} \tag{4.34}$$

$$Q_3(t_{k+1}) = Q_3(t_k) + TI_P - \frac{T^2 C_2 I_P - TC_2C_3\left(\frac{Q_2(t_k)}{C_2} - \frac{Q_3(t_k)}{C_3}\right)}{T(C_2+C_3) + R_2C_2C_3} \tag{4.35}$$

$$V_C(t_{k+1}) = \frac{Q_3(t_{k+1})}{C_3} \tag{4.36}$$

Again it is important to emphasise that while the above equations are involved, the derivation contained no differential or integral terms. Thus, they are solvable. The error introduced due to the charge approximation is directly proportional to T and is considered in more detail in Section 4.3.5.

4.3.3 Fourth Order DPLL Charge Approximation

To derive the fourth order DPLL charge approximation the same steps as used in Section 4.2.1 earlier will be used. The derivation is as follows:

1. Assign values to all the positive system nodes (V_A, V_B, and V_D) in the third order filter as in Figure 4.10 below.

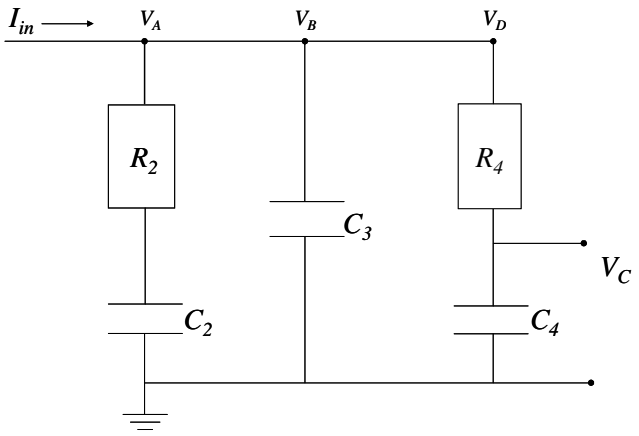

Figure 4.10 Assignment of Third Order Filter Nodes

Next determine the voltage across each section in terms of the capacitor charge as in equation (4.37 – 4.39).

$$V_A(t_{k+1}) = I_2(t_{k+1})R_2 + \frac{Q_2(t_k) + \overline{I_2}T}{C_2} \tag{4.37}$$

$$V_B(t_{k+1}) = \frac{Q_3(t_k) + \overline{I_3}T}{C_3} \tag{4.38}$$

$$V_D(t_{k+1}) = I_4(t_{k+1})R_4 + \frac{Q_4(t_k) + \overline{I_4}T}{C_4} \tag{4.39}$$

2. Equate each of the node voltages V_A=V_B, V_A=V_D and V_B=V_D as in equations (4.40, 4.41, and 4.42) respectively and add the current into the node expression as in equation (4.43).

$$I_2(t_{k+1})R_2 + \frac{Q_2(t_k) + \overline{I_2}T}{C_2} = \frac{Q_3(t_k) + \overline{I_3}T}{C_3} \tag{4.40}$$

$$I_2(t_{k+1})R_2 + \frac{Q_2(t_k) + \overline{I_2}T}{C_2} = I_4(t_{k+1})R_4 + \frac{Q_4(t_k) + \overline{I_4}T}{C_4} \tag{4.41}$$

$$I_4(t_{k+1})R_4 + \frac{Q_4(t_k) + \overline{I_4}T}{C_4} = \frac{Q_3(t_k) + \overline{I_3}T}{C_3} \tag{4.42}$$

$$I_{in} = I_2(t_{k+1}) + I_3(t_{k+1}) + I_4(t_{k+1}) \tag{4.43}$$

3. Approximating the average currents in equations (4.40 – 4.43) to $I(t_{k+1})$ the current at time t_{k+1}, we are left with four simultaneous equations and three unknowns ($I_2(t_{k+1})$, $I_3(t_{k+1})$ and $I_4(t_{k+1})$). Again remember that I_{in} is the current into the filter which may be $+I_p$, $-I_p$ or 0, depending on the PFD state.

4. The next step is to solve these simultaneous equations for each current expression as in equations (4.44 – 4.46) below.

$$I_2(t_{k+1}) = \frac{I_4(t_{k+1})\left(C_2C_4R_4 + TC_2\right) - C_2C_4\rho_2}{R_2C_2C_4 + TC_4} \tag{4.44}$$

$$I_4(t_{k+1}) = \frac{I_3(t_{k+1})TC_4 + C_3C_4\rho_3}{R_4C_3C_4 + TC_3} \tag{4.45}$$

$$I_3(t_{k+1}) = \frac{I_{in} + \dfrac{\rho_1}{R_2 + \dfrac{T}{C_2}} - \dfrac{\rho_3}{R_4 + \dfrac{T}{C_4}}}{\dfrac{TC_4}{R_4C_3C_4 + TC_3} + \dfrac{TC_2}{R_2C_2C_3 + TC_3} + 1} \tag{4.46}$$

where $\rho_1 = \dfrac{Q_2(t_k)}{C_2} - \dfrac{Q_3(t_k)}{C_3}$, $\rho_2 = \dfrac{Q_2(t_k)}{C_2} - \dfrac{Q_4(t_k)}{C_4}$ and $\rho_3 = \dfrac{Q_3(t_k)}{C_3} - \dfrac{Q_4(t_k)}{C_4}$.

5 Now that $I_3(t_{k+1})$ is expressed in terms of known values, it is possible to calculate all the current values, the charge on each filter capacitor, and the filter output voltage using equations (4.47 – 4.51) as given below.

$$I_2(t_{k+1}) = I_{in} - I_3(t_{k+1}) - \frac{I_3(t_{k+1})TC_4 + C_3C_4\rho_3}{R_4C_3C_4 + TC_3} \tag{4.47}$$

$$I_4(t_{k+1}) = I_{in} - I_2(t_{k+1}) - I_3(t_{k+1}) \tag{4.48}$$

$$Q_2(t_{k+1}) = Q_2(t_k) + T\left(\frac{I_2(t_{k+1}) + I_2(t_k)}{2}\right) \tag{4.49}$$

$$Q_3(t_{k+1}) = Q_3(t_k) + T\left(\frac{I_3(t_{k+1}) + I_3(t_k)}{2}\right) \tag{4.50}$$

$$Q_4(t_{k+1}) = Q_4(t_k) + T\left(\frac{I_4(t_{k+1}) + I_4(t_k)}{2}\right) \tag{4.51}$$

(Note that in equations (4.49 – 4.51) ΔQ is calculated without the use of the charge approximation as the currents at time (t_{k+1}) are known in this case). Finally V_C can be estimated using:

$$V_C(t_{k+1}) = \frac{Q_4(t_{k+1})}{C_4} \tag{4.52}$$

The above set of equations from (4.47) up to (4.52) can be reduced to the set of equations (4.53 – 4.56), these are the charge approximated equations for the fourth order DPLL.

$$Q_2(t_{k+1}) = Q_2(t_k) + \frac{T_P^2 I_3 C_2 - T_P C_2 C_3 (\frac{Q_2(t_k)}{C_2} - \frac{Q_3(t_k)}{C_3})}{T_P C_3 + R_2 C_2 C_3} \tag{4.53}$$

$$Q_3(t_{k+1}) = Q_3(t_k) + T_P I_3 \tag{4.54}$$

$$Q_4(t_{k+1}) = Q_4(t_k) + \frac{T_P^2 I_3 C_2 - T_P C_2 C_3 (\frac{Q_2(t_k)}{C_2} - \frac{Q_3(t_k)}{C_3})}{T_P C_3 + R_2 C_2 C_3} + T_P I_3 - T_P I_P \tag{4.55}$$

$$V_C(t_{k+1}) = \frac{Q_4(t_{k+1})}{C_4} \tag{4.56}$$

where I_3 is the current through C_3 and is calculated as in equation (4.57), and T_P is the period of the current state, i.e. it is equal to T_B in the boost state[vi] and T_C in the coast state.

[vi] Note that the boost and coast state terminology is being introduced here. The boost state corresponds to either the Up or Down state of the CP-PFD where there is a constant current being

$$I_3 = \frac{\dfrac{\left(\dfrac{Q_2(t_k)}{C_2} - \dfrac{Q_3(t_k)}{C_3}\right)}{R_2 + \dfrac{T_P}{C_2}} - \dfrac{\left(\dfrac{Q_3(t_k)}{C_3} - \dfrac{Q_4(t_k)}{C_4}\right)}{R_4 + \dfrac{T_P}{C_4}} - I_p}{\dfrac{C_2 T_P}{C_3 T_p + C_2 C_3 R_2} + \dfrac{C_4 T_P}{C_3 T_p + C_3 C_4 R_4} + 1} \tag{4.57}$$

Using these equations and the phase signals in equations (4.29) and (4.30) from earlier, the fourth order DPLL system can be modelled.

4.3.4 Fifth Order DPLL Charge Approximation

The fifth order DPLL behavioural equations are derived using the same approach as applied to the previous orders. The derivation in this case is not included here but the resulting fifth order charge approximated system model equations are as given in equations (4.58 – 4.65) below.

$$Q_2(t_{k+1}) = Q_2(t_k) + I_2 T_P \tag{4.58}$$

$$Q_3(t_{k+1}) = Q_3(t_k) + \frac{I_2 C_3 (R_2 C_2 + T) + C_3 Q_2(t_k) - C_2 Q_3(t_k)}{C_2 T} \tag{4.59}$$

$$V_C(t_{k+1}) = \frac{Q_5(t_{k+1})}{C_5} \tag{4.60}$$

where I_2 and D_1 to D_4 are defined as given in equations (4.61 – 4.65).

---

pumped into the loop filter (either $+I_p$ or $-I_p$ A). The coast state corresponds to the Null state of the CP-PFD where no current is being pumped into the loop filter.

$$I_2 = \frac{D_3 - D_4\rho_1}{1+(R_2C_2+T)\dfrac{D_4}{C_2}} \tag{4.61}$$

$$D_1 = C_5R_4\rho_4 - \rho_3(R_5C_5+T) \tag{4.62}$$

$$D_2 = \frac{(R_4C_4+T)(R_3C_3+T)+C_5R_4T}{C_4} \tag{4.63}$$

$$D_3 = \frac{D_1(C_4(R_5C_5+T)+C_5T)}{D_2C_4(R_5C_5+T)} - I_P + \frac{C_5\rho_3}{R_5C_5+T} \tag{4.64}$$

$$D_4 = \frac{C_3C_4T+TC_4(R_5C_5+T)+C_5T^2}{C_4TD_2} \tag{4.65}$$

where $\rho_4 = \left(\frac{Q_4(t_k)}{C_4} - \frac{Q_5(t_k)}{C_5}\right)$.

Again similar to the previous cases, by using these set of equations and the phase equations in (4.29) and (4.30), the fifth order DPLL system can be modelled. It is worth noting here that the fifth order DPLL behavioural equations can only be determined by making this charge approximation, and cannot be derived using existing methods.

4.3.5 DPLL Charge Approximation Limitations and Error

The major benefit of the charge approximation is that behavioural equations for high order DPLL systems can be derived. This is not achievable using existing behavioural methods. In theory there is no restriction on the order of the DPLL system that can be modelled.

The error introduced by the charge approximation has already been discussed in the case of a single capacitor and a second order filter. In terms of the second order filter it was concluded that the approximation error can be justified if the current source supplied to the filter is for only a short period of the filter time constant. This is the case for high frequency mixed signal devices. Consider a DPLL system, if the reference signal has a frequency, F_R, of 200MHz then the sample time T is equal to

5ns, this means that the loop filter will only charge continuously for a fraction of this time period, T_B, when the PFD is in the Up or Down states (boost state). The length of time that the PFD is in the boost state, T_B, can be calculated from the absolute value of the feedback loop phase error signal[vii] ϕ_e which is equal to ϕ_R - ϕ_V from equations (4.29) and (4.30). The exact calculation of the boost period is given in below.

$$T_B = \left| \frac{\phi_e(t_k)T}{2\pi} \right| \tag{4.66}$$

Thus the boost period (and therefore the charge approximation error) is going to vary depending on ϕ_e and the reference frequency (as $T = 1/F_R$). The relationship between the approximation error and ϕ_e is important. As ϕ_e approaches zero (as the loop approaches lock), the approximation error approaches zero. This is significant as the approximation error is a reducing error that is already bounded by a small time period T_B. This is illustrated in Figure 4.11 below, which shows that the accuracy improves as we approach lock and as the reference frequency is increased.

[vii] The absolute value of the phase error is used because the sign of ϕ_e corresponds to the state of the PFD (i.e. positive in the Up state and negative in the Down state), the sign is not important if a measure of the sample time is required. What is important is the absolute magnitude of ϕ_e.

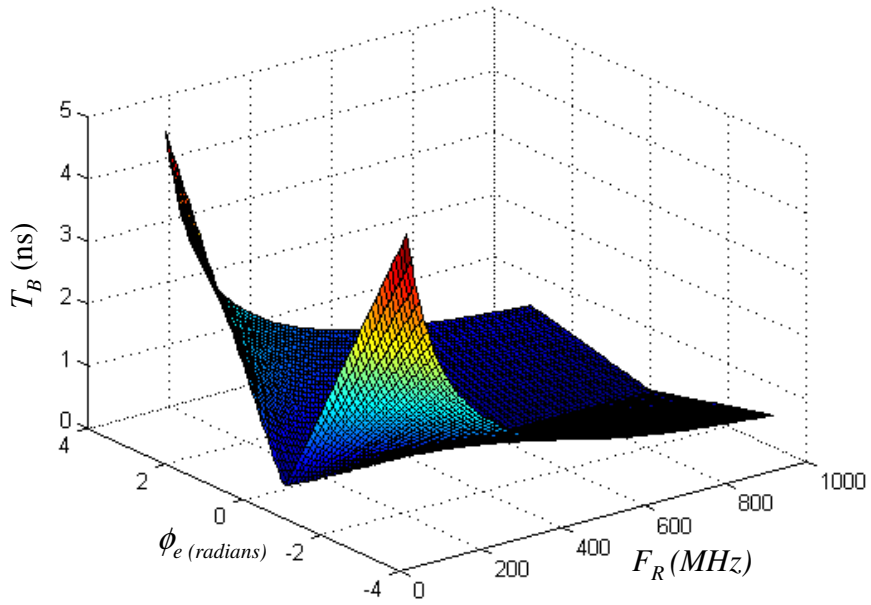

Figure 4.11 Estimation of the Boost Period for a Typical DPLL system

Thus the error is inversely proportional to the reference frequency F_R, directly proportional to the time interval T, and directly proportional to the loop phase error (or the closeness to loop lock).

Finally to compare the charge approximated behavioural equations of the DPLL relative to existing behavioural methods [50], consider as an example a third order DPLL system with a set of system parameters given as R_2 = 16kΩ; C_2 = 200pF; C_3 = 100pF; I_P = 10μA; K_V = 30x10^6 Hz/V; T = 0.3ns; N = 1. The transient response of the behavioural model for this system is plotted in Figure 4.12 below and the system response of the proposed charge approximated model is plotted in Figure 4.13.

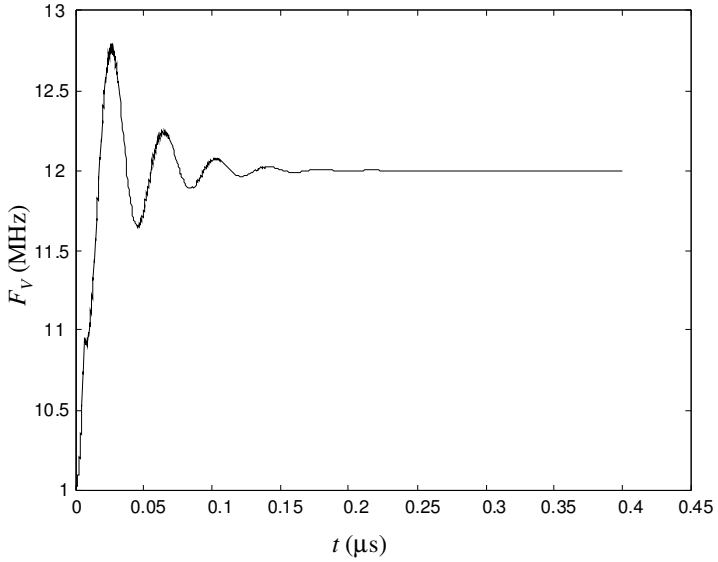

Figure 4.12 Frequency Response of an Existing Behavioural DPLL System Model [50]

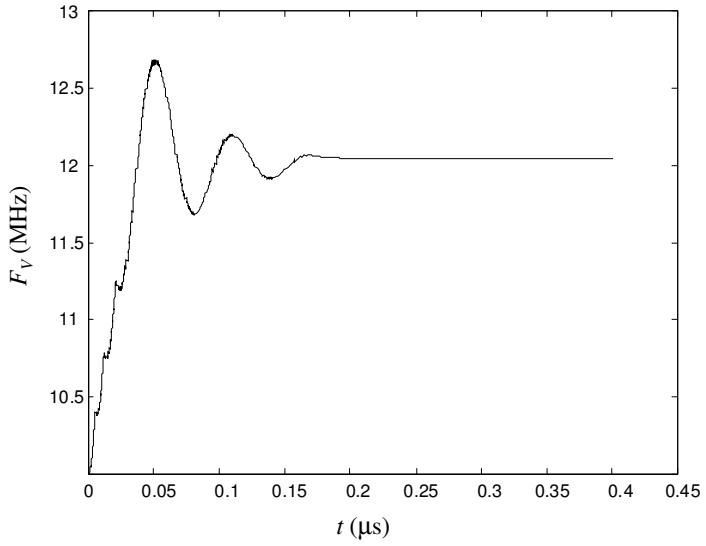

Figure 4.13 Frequency Response of the Equivalent Charge Approximation Model

The error between these two signals, for a range of T, is given in Figure 4.14. It can be seen that the error introduced by the charge approximation, is reduced as value of the time step is decreased.

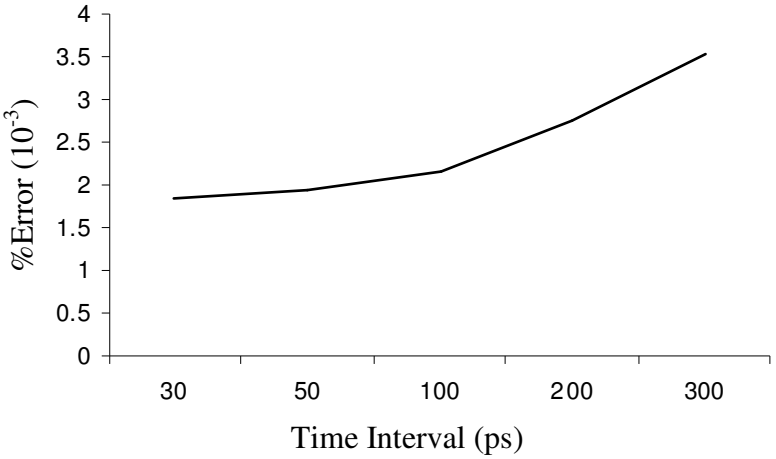

Figure 4.14 Plot Showing Decreasing Error as *T* is Reduced

The selection of *T* introduces a trade off between the error and the simulation time. As *T* reduces, the error reduces correspondingly, with an increase in the simulation time as shown in Figure 4.15.

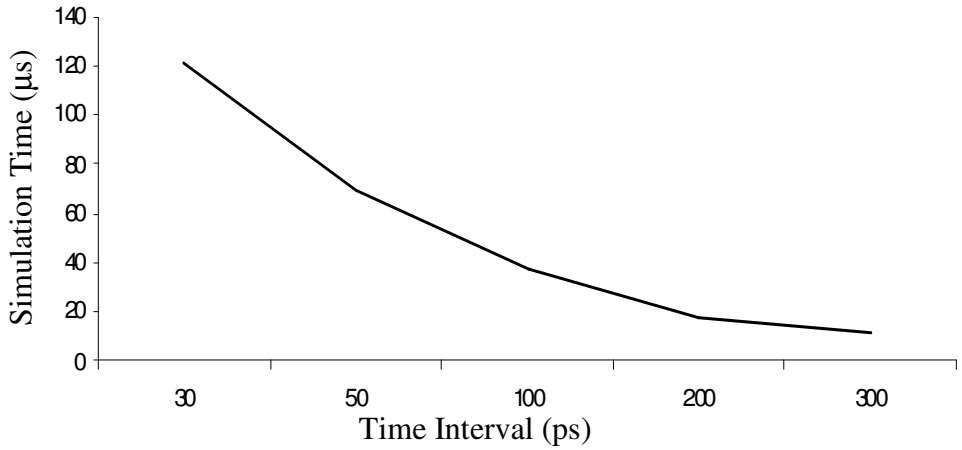

Figure 4.15 Plot of Decreasing Simulation Time as the Step Size is Increased

4.4 Conclusion

The charge approximation methodology proposed in this chapter is a significant improvement over existing mixed signal behavioural modelling methods in that it results in less complex system model equations, with no differential terms. To illustrate this, a simple RC transconductance filter was used. It was shown that by employing this approximation the differential terms in the filter equations can be

removed, with the significant benefit that the complexity of high order system equations is reduced, thus the restriction on the system order is removed.

The charge approximation, however, is introduced at the expense of some additional modelling error that is directly proportional to the simulation time interval. It was found that the error introduced by the approximations made in this model are bounded by the choice of Δt, the period, hence it introduces a trade off between simulation time and accuracy. The approximation error can be justified for high order systems by considering that if a constant current I_{in} is applied to the filter input for a short period of time, i.e. a fraction of Δt, then the filter capacitor will charge slowly, thus changing by only a small amount. In the case of high frequency DPLL systems, the filter input is a narrow current pulse, thus the change in the filter capacitor charge is minimal and the response can be accurately modelled with the first order linear charge approximation as described. It was also found that the charge approximation reduces as the feedback loop phase error approaches zero and as the loop approaches lock.

As a particular design example the third order DPLL system was considered in detail in Section 4.3. It was shown that the system could be accurately modelled using the charge approximation methodology. The charge approximation methodology was used here to accurately model fourth and fifth order DPLL systems, something that is not possible with existing behavioural models. However in theory there is no restriction on the model order. While the application of the charge approximation methodology to the DPLL system is considered in detail in this thesis, the charge approximation also offers significant potential to the high order behavioural modelling of other mixed signal devices, however this is beyond the scope of this thesis.

CHAPTER 5

STABILITY BOUNDARIES OF HIGH ORDER DPLL SYSTEMS USING PIECEWISE LINEAR METHODS

5.1 Introduction

Stability boundaries are commonly used as an initial step in the PLL design process. An example of a stability boundary (for the second order PLL) is given in Figure 3.17 earlier. This plot defines the stability of the PLL of a range of system gains and loop filter time constants. By choosing a set of system parameters with a system gain and a loop filter time constant within the stable region of Figure 3.17 a stable system can be designed. This initial design step is still used to good effect today. In the case of high order DPLL systems the system model equations become increasingly complex. As a result the linear prediction of the system boundaries can not be determined. Thus high order stability boundaries are unusable which means there is no linear design start point as in the case of low orders. As a result of this high order DPLL design is particularly time consuming and is considered comparatively risky due to the increased loop complexity.

In this chapter the high order DPLL stability boundaries restriction is overcome by using a novel piecewise linear methodology. This methodology utilises the charge approximation presented in the previous chapter thereby removing the high order restriction and enables the determination of high order stability boundaries without mathematical restrictions. The proposed piecewise linear stability methodology starts with the charge approximated equations from the previous chapter. Using these, the response of the DPLL state variables are determined over a short period of time, using a small number of piecewise linear steps. From this prediction of the state space response it is possible to make an early prediction of the loop stability and lock time. This stability methodology is presented in more detail in the next subsection.

There are currently two ways to determine the DPLL stability boundary: the first method is to use the linear PLL approximation (as in Figure 3.17); and the second method is to simulate a set of points in the stability plane using a circuit level simulation, from the simulation results the boundary can be plotted as a line between stable and unstable simulation points. In this chapter both boundaries will be used as a reference for the results of the new proposed methodology; the linear boundary is the reference to existing methods; the circuit level method being the 'real' stability boundary of the system. The stability boundary methodology is presented in Section 5.2; it is shown to significantly improve the accuracy of the stability prediction (with the additional benefit of reducing the simulation time relative to the circuit level prediction).

In Section 5.3 the proposed piecewise linear method is applied to the second order DPLL system. The stability boundary for this system is found using a closed form solution which is then used to determine the global stability boundary of the second order system. The new stability boundary is then compared to the linear boundary and the circuit level simulation derived boundary. The closed form solution to the second order stability boundary is made possible because of the new piecewise linear technique proposed in this section. However higher order stability boundaries cannot be determined in closed form but can be determined by iterating over the equivalent high order piecewise linear system equations. The determination of such high order stability boundaries is considered in Section 5.4.

5.2 Piecewise Linear model

The definition of stability for a feedback system is that for any bounded input to this system, for any period of time, the output will also be bounded. To a classical control theorist this means that a feedback system is stable if the feedback loop error signal approaches and settles to some equilibrium point within a predefined period of time. To a PLL designer on the other hand, stability is when the system loop error approaches an equilibrium point and stays within a predefined region (i.e. the tracking region plotted in Figure 2.5 earlier) about the equilibrium for a period of time. In other words the PLL can oscillate about the equilibrium and not settle to it, as is required for classical stability, see Figure 5.1 below. This definition of PLL stability is known as 'loop lock'.

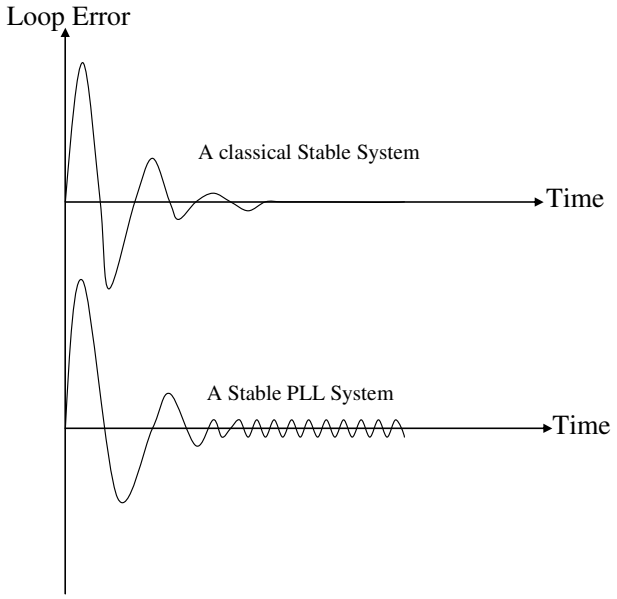

Figure 5.1 Stability Definitions

If the classical definition of stability was applied to the DPLL, then it would be an arduous task to design a stable DPLL. The reason for this is to do with the nature of the DPLL, the loop in reality can never be phase and frequency locked with zero phase error. When locked even the most optimally designed DPLL systems will have some phase error offset due to jitter in the reference and VCO signals; loop component noise and/or intrinsic changes in loop parameters (such as the feedback divide ratio). When the loop loses lock, for whatever reason, it is designed to restore

lock as quickly as possible. Fast lock is achieved by two specific means: the choice of PD (a charge pump PFD has a faster lock performance then the mixing PD); or the choice of loop bandwidth. A wide bandwidth system will lock faster but this is at the expense of a greater phase error offset.

A more intuitive way of considering the system stability is in state space [60, 61]. If all state variables are decreasing overtime, i.e. approaching some equilibrium, then the stability of the system can be determined by viewing the response of each state variable to a small offset for a short period of time. This approach is similar to that of Lyapunov's direct method described earlier in Chapter 3, which states that state variables in a stable system which start close to an equilibrium will remain close to the equilibrium, and eventually converging to the equilibrium. In this chapter a novel means of determining the system stability by monitoring the trajectory of the state variables in state space is presented.

Consider a second order DPLL system, such a system has two dependent state variables, the phase error ϕ_e and the VCO control voltage V_C. If a stable system is initialised with a phase error of zero radians, a small control voltage offset of V_0 volts and a VCO free running frequency F_{FR} equal to the reference frequency, then the expected response is that the state space trajectory would be in an anti-clockwise direction spiralling into the origin. This state space response is plotted in Figure 5.2 below.

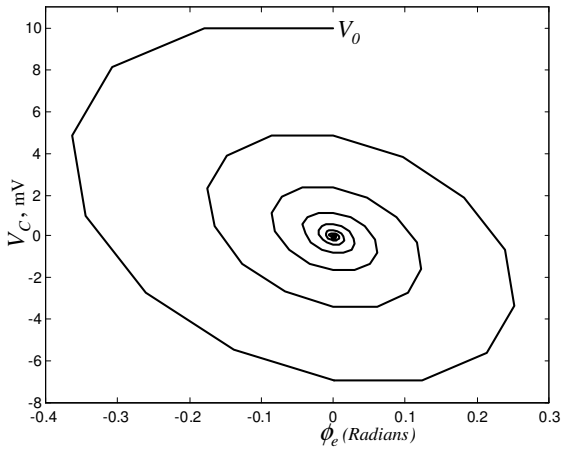

Figure 5.2 State Space Trajectory of Stable System

Looking at the system trajectory of Figure 5.2 it can be concluded that this particular DPLL system is stable. By plotting the trajectory of any DPLL system it can be seen whether that particular system is stable or unstable (i.e. if the trajectory converges or diverges respectively). The determination of the system stability from the state space trajectory can be further simplified by just considering left hand segment of the state space trajectory, i.e. the solid line in Figure 5.3.

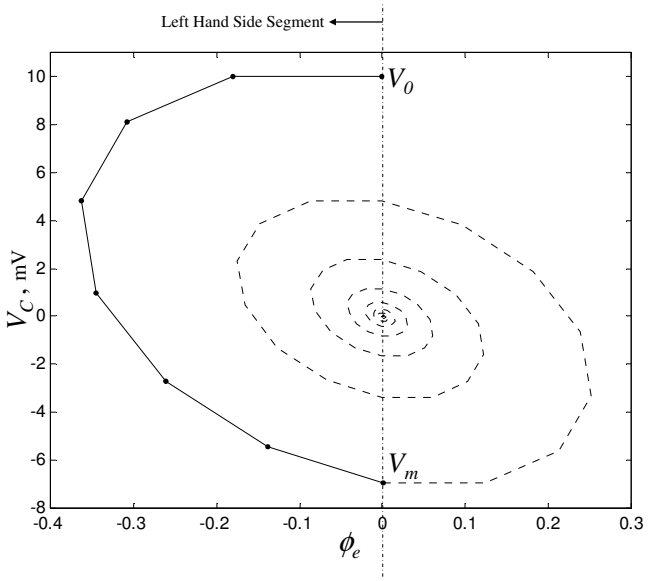

Figure 5.3 Left Hand Segment of the State Space Trajectory

The solid line in Figure 5.3 is a piecewise linear curve where the dots correspond to samples of the VCO control voltage. In the case of the second order DPLL, the system is linear between these sample points, allowing the piecewise linear method to give an exact calculation of this state space curve. This method assumes a small initial VCO control voltage offset V_0 and determines the system stability from the state space response to this offset. This offset voltage V_0 is chosen to be small for two reasons:

1. It has already been stated that this technique utilizes that charge approximated DPLL behavioural equations. From the analysis in the previous chapter it is known that the charge approximation error is proportional to the phase error offset, this phase error is directly related to V_0, so to minimise the error a small V_0 is required.

2. Second a small V_0 ensures that the maximum phase offset remains within the +/- 2π region, avoiding cycle slip events. As discussed in Section 2.5 earlier, cycle slips occur when the feedback signal, to which the reference signal is being compared, changes instantaneously by a large amount (due to noise) incurring an immediate shift in the phase error of greater than 2π radians. Cycle slip events can be explained by this analysis but are beyond the scope of this thesis.

It will be shown in section 5.3.2 that the stability boundary is independent of the initial control voltage offset V_0, this is significant for the following reason. When the system trajectory crosses the phase error zero crossing (i.e. ϕ_e goes from negative to positive) the PFD inputs have changed from the VCO signal leading the reference signal to lagging it. If it can proven that this trajectory is independent of the initial control voltage offset V_0, then it can be concluded that this half segment accurately represents any other half segment of this system in state space. In other words, if this segment is converging to the zero equilibrium, then all other segments are. Thus the overall system is converging and is therefore stable. This proof will be considered in section 5.3.2 later when the piecewise linear system equations have been derived.

The state trajectory of Figure 5.3 can be used to define the system stability by considering the transition values of V_m and V_0. If V_m, which is the first zero crossing of ϕ_e, can be calculated, then the system stability can be determined as follows: if $|V_m| < V_0$[viii] then the trajectory is converging and is stable, see Table 5.1; otherwise if $|V_m| > V_0$ then the system trajectory is diverging and is therefore unstable, as in Figure 5.4 below.

[viii] Assuming a positive V_0 value

System Performance	Condition
Unstable	$\|V_m\| \geq V_0$
Stable	$\|V_m\| < V_0$

Table 5.1 Stability Criterion

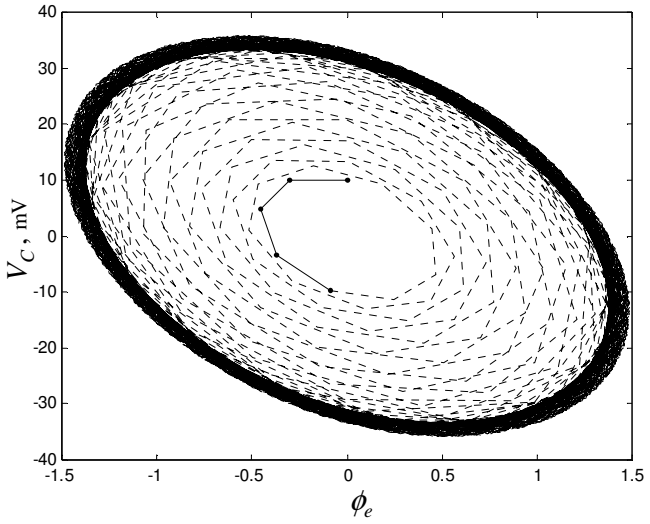

Figure 5.4 State Space Trajectory of an Unstable System

The above stability criterion will only be accurate if a good approximation of V_m can be found. The more piecewise linear state space sample points that exist in the left half plane the more accurate the prediction of V_m will be. If too few sample points exist, let's say one sample point, then the system trajectory will not be accurately determined, see Figure 5.5 below.

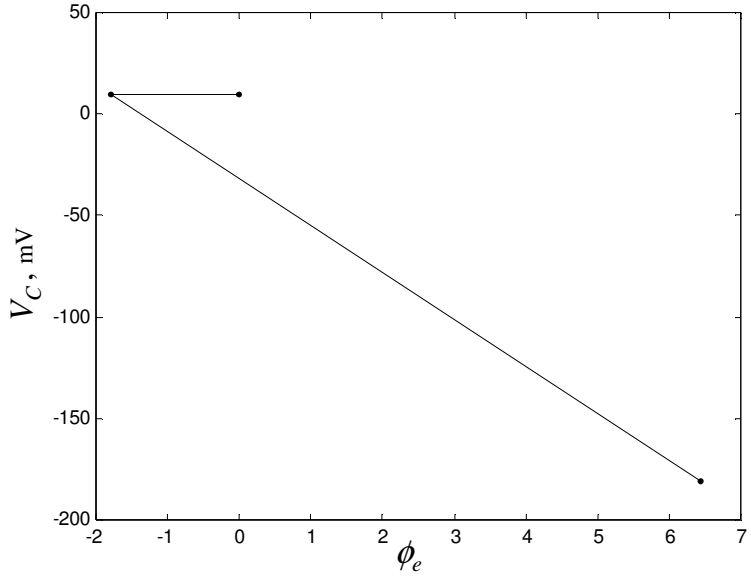

Figure 5.5 Plot of System Trajectory with too few Sample Points

In the example of Figure 5.5 the estimation of V_m is inaccurate because there is only one state space sample point in the left hand segment of the state space plane, and therefore does not accurately interpret the system trajectory. The number of state space sample points in one segment of the trajectory is directly proportional to the reference frequency. Thus the accuracy of this piecewise linear methodology increases with increasing reference frequency. An estimate of the number of sample points on this trajectory, m, is determined in section 5.3.3.

In the next section the piecewise linear methodology will be used to derive the second order DPLL stability boundary in closed form. This second order solution is then compared to the linear and circuit level derived boundaries. Finally in Section 5.2 the proposed methodology is applied to the third and fourth order DPLL systems. From these results the benefits and limitations of the proposed technique are considered.

5.3 Second Order Closed Form DPLL Stability Boundary Estimation

In this section the stability boundary of the second order DPLL system is determined using the piecewise linear method as introduced in the previous section. This stability methodology determines the control voltage V_C after a number of periods of the reference signal. The plot of the state space system trajectory (given in Figure 5.3) can be determined by plotting the two state variables, V_C and ϕ_e, against each other. For a stable system with a reference frequency equal to the VCO free running frequency F_{FR}, and an initial control voltage offset of V_0 volts, the system will settle to the equilibrium of the origin as shown earlier in Figure 5.3. If the value of V_m can be estimated, which is the control voltage when the phase error reaches zero for the first time after time $t_k = 0$, then the system stability can be determined using the criteria of Table 5.1.

In this section the second order DPLL control voltage after m periods of the reference signal, V_m, as defined in Figure 5.3, is determined. The control voltage expression, V_m, is then used to define a closed form solution of the DPLL stability boundary for the second order system. To determine V_m two things need to be considered: first, the n^{th} period of the control voltage V_n needs to be calculated in closed form; and second the number of periods m needs to be calculated where V_m is the control voltage at the first zero crossing of the phase error as shown in Figure 5.3. The first requirement is to enable the calculation of any of the state space sample points in Figure 5.3; the second requirement is to enable the approximation of V_m, which may lie between two sample points. The calculation of both parameters is considered in the following section and later in Section 5.3.3.

5.3.1 Calculation of the Control Voltage Sample Point V_n

Using the charge approximate DPLL behavioural equations derived in Chapter 4, an estimate of the two state variables $\phi_e(t_k)$ and $V_C(t_k)$ can be determined, where t_k is the time at the k^{th} period of the reference signal. Note that the sample period is equal to one period of the reference signal, i.e. $1/F_R$. However in the case of high order

systems the accuracy of the model can be increased by reducing this sample period to some fraction of the reference signal period, this is discussed in more detail section 5.4. In the case of the second order DPLL system response, the state variables can be described using the set of behavioural equations (5.1) and (5.2) given below.

$$V_C(t_{k+1}) = V_C(t_k) - \frac{T_B I_{in}}{C_2} \tag{5.1}$$

$$\phi_e(t_{k+1}) = \phi_e(t_k) + 2\pi\left[\left(F_R - F_{FR}\right)T - K_V \int V_C dt\right] \tag{5.2}$$

where

$$\int V_C dt = TV_C(t_k) - T_B I_{in} R_2 - \frac{T_B^2 I_{in}}{2C_2} \tag{5.3}$$

The control voltage behavioural equation (5.1) can be explained by considering one time period, T, of the DPLL loop as in Figure 5.6.

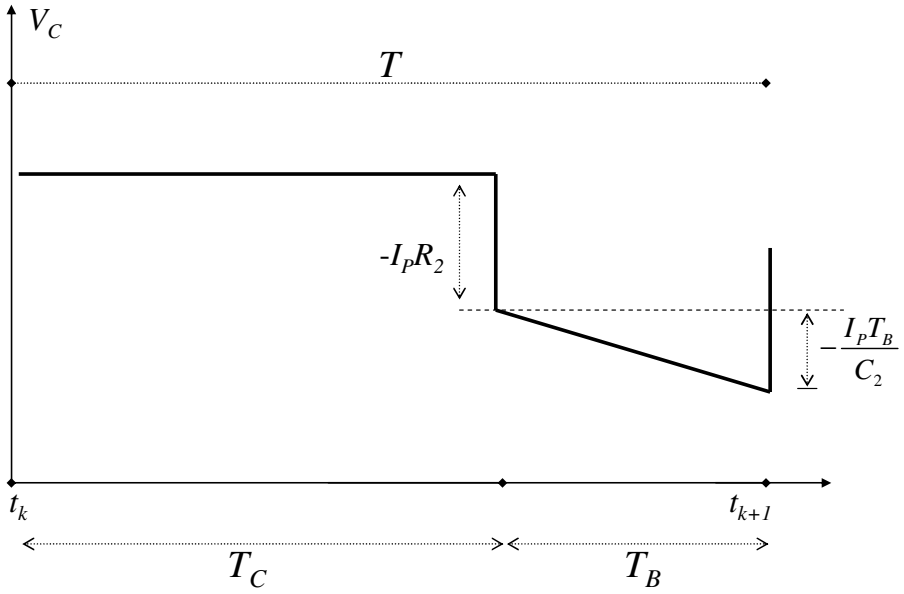

Figure 5.6 One Period of Control Voltage for the Second Order DPLL

The DPLL loop operates in two of the system states during the time period, T, these states are: the coast state, where no current is output from the CP-PFD component, for a period of time, T_C; and in boost state for a period of T_B.

If a positive initial control voltage offset of V_0 is applied, the VCO signal will initially lead the reference, thus the PFD output is expected to be a negative current

pulse of $-I_P$ amps. This negative current pulse will persist in the loop while the system state trajectory remains in the left half hemisphere of state space. Once the trajectory moves into the right half plane, the CP-PFD will output a positive current pulse during the boost period. This methodology only requires the determination of the system trajectory in the left half plane of state space, and thus the simplification is that the output of the CP-PFD is 0 amps for T_C seconds and $-I_P$ amps for T_B seconds. From this equations (5.1) and (5.3) can be rewritten as in equations (5.4 – 5.5) below.

$$V_C(t_{k+1}) = V_C(t_k) - \frac{T_B I_P}{C_2} \tag{5.4}$$

$$\int V_C dt = TV_C(t_k) - T_B I_P R_2 - \frac{T_B^2 I_P}{2C_2} \tag{5.5}$$

To calculate T_B, consider that T_B is some fraction of the total time period T. This fraction is defined by the phase error divided by the total maximum possible error. Thus T_B can be expressed by equation (5.6).

$$T_B = T\frac{\left|\phi_e(t_k)\right|}{2\pi} \tag{5.6}$$

If the state space origin (i.e. zero phase error and zero control voltage) is defined as the equilibrium, then we require that the reference frequency F_R be equal to the free running frequency F_{FR}. Substituting equation (5.5) into (5.2) we can rewrite the phase error at period t_k as:

$$\phi_e(t_{k+1}) = \phi_e(t_k) - 2\pi K_V\left[TV_C(t_k) - T_B I_P R_2 - \frac{T_B^2 I_P}{2C_2}\right] \tag{5.7}$$

Given that the boost time T_B is given in equation (5.6), and substituting it into (5.7), we get the phase error behavioural equation as in equation (5.8).

$$\phi_e(t_{k+1}) = \phi_e(t_k) - 2\pi K_V\left[TV_C(t_k) - \frac{TI_P R_2\left|\phi_e(t_k)\right|}{2\pi} - \frac{T^2 I_P \phi_e^2(t_k)}{8\pi^2 C_2}\right] \tag{5.8}$$

As both T^2 and $\phi_e^2(t_k)$ are small numbers equation (5.8) can be approximated by equation (5.9). The error introduced is directly related to the square of the time

period T and the square of the phase error. Both of these parameters have already been kept small, by choosing a small V_0 and considering only high frequency systems.

$$\phi_e(t_{k+1}) \approx \phi_e(t_k) - 2\pi K_V \left[TV_C(t_k) - \frac{TI_P R_2 \left|\phi_e(t_k)\right|}{2\pi} \right] \tag{5.9}$$

As the phase error ϕ_e is always negative, for the state space trajectory segment under analysis, then it is the case that $|\phi_e|$ is equal to $-\phi_e$ and therefore equation (5.9) can be simplified as:

$$\phi_e(t_{k+1}) \approx \phi_e(t_k)\left[1 - K_V TI_P R_2\right] - 2\pi K_V TV_C(t_k) \tag{5.10}$$

Finally by examination of the first few iterations of equation (5.10), it is possible to determine the feedback loop phase error at sample time j in terms of the control voltage. The first four iterations are given in Table 5.2 below, where $B = 1\text{-}K_V TI_P R_2$, $D = -2\pi K_V T$, and V_1 corresponds to $V_C(1)$.

Iteration	Phase Error
1	$\phi_e(1) = DV_0$
2	$\phi_e(2) = BDV_0 + DV_1$
3	$\phi_e(3) = B^2 DV_0 + BDV_1 + DV_2$
4	$\phi_e(4) = B^3 DV_0 + B^2 DV_1 + BDV_2 + DV_3$

Table 5.2 First Four Iterations of the Phase Error

By inspection of the first four iterations of ϕ_e it can be concluded that equation (5.10) can be solved using equation (5.11), this is the feedback loop phase error at sample point j.

$$\phi_e(j) = -2\pi K_V T \sum_{i=0}^{j-1} \left(1 - TK_V I_P R_2\right)^{j-i-1} V_C(i) \tag{5.11}$$

In a similar manner the control voltage at sample point *j* can also be found by taking the charge approximated control voltage behavioural equation, (5.4) and substituting in T_B from equation (5.6), the control voltage can be rewritten as:

$$V_C(t_{k+1}) = V_C(t_k) - \frac{I_P}{C_2}\frac{T}{2\pi}\left|\phi_e(t_k)\right| \tag{5.12}$$

Again let $|\phi_e|$ equal to $-\phi_e$, therefore the control voltage becomes:

$$V_C(t_{k+1}) = V_C(t_k) + \frac{TI_P\phi_e(t_k)}{2\pi C_2} \tag{5.13}$$

Finally by looking at the first few iterations of equation (5.13), see Table 5.3 (where $F = I_PT/2\pi C_2$), the control voltage at sample point *j* can be expressed as in equation (5.14).

Iteration	**Control Voltage**
1	$V_C(1) = V_0 + F\phi_0 = V_0$
2	$V_C(2) = V_0 + F\phi_1$
3	$V_C(3) = V_0 + F\phi_1 + F\phi_2$
4	$V_C(4) = V_0 + F\phi_1 + F\phi_2 + F\phi_3$

Table 5.3 First Four Iterations of the Control Voltage

$$V_C(j) = V_0 + \frac{I_PT}{2\pi C_2}\sum_{i=0}^{j-1}\phi_e(i) \tag{5.14}$$

Equations (5.11) and (5.14) can both be used to determine the response of the system state variables to a small control voltage offset V_0, with an initial ϕ_e offset of zero, and are valid while the phase error remains negative (i.e. while the system trajectory remains in the left hemisphere of state space as in Figure 5.3).

Equations (5.11) and (5.14) can be reduced to one expression by substituting equation (5.11) into (5.14). The resulting double summation expression in equation (5.15) is an estimate of the control voltage V_n and is independent of the phase.

$$V_n = V_0 - \frac{K_V I_P T^2}{C_2} \sum_{i=0}^{n-1} \sum_{k=0}^{i-1} \left(1 - K_V I_P R_2 T\right)^{i-j-1} V_C(k) \tag{5.15}$$

where V_n is equivalent to $V_C(n)$ and is an estimate of the control voltage after n periods of the reference signal.

This estimate of V_n will be valuable in the determination of the stability of the DPLL system as it is required to estimate the control voltage at the phase error zero crossing (V_m in Figure 5.3). In the remainder of this section equation (5.15) will be used to prove that the stability boundary is independent of the initial control voltage offset V_0.

5.3.2 Proof that the Stability Boundary is Independent of Initial Conditions

In this section it is shown that the piecewise linear stability criterion defined earlier is independent of the initial control voltage offset V_0. If the defined stability prediction is independent of V_0, then the stability prediction is a global prediction of the particular system for all possible voltage offsets; thus the system is stable for all voltage offsets regardless of the source (due to noise or changing system parameters).

It has already been discussed that the system stability can be determined by the value of V_m from Table 5.1; i.e. if the absolute value of V_m is less than V_0 the system is stable, otherwise the system is unstable. Thus it can be concluded that the system is stable where:

$$V_0 > |V_m| \tag{5.16}$$

$$\Rightarrow V_0 - |V_m| > 0 \tag{5.17}$$

Using this definition of stability and letting the solution to the control voltage V_n in equation (5.15), equal to V_m and substituting into equation (5.17) yields:

$$V_0 - \left| V_0 - \frac{K_V I_P T^2}{C_2} \sum_{i=0}^{n-1} \sum_{k=0}^{i-1} \left(1 - K_V I_P R_2 T\right)^{i-j-1} V_C(k) \right| > 0 \tag{5.18}$$

Let Y equal the large double summation expression of equation (5.18), as in equation (5.19).

$$V_0 - |V_0 - Y| > 0 \tag{5.19}$$

By making the following set of observations, it can be concluded that Y is some positive value:

1 The initial control voltage offset V_0 is always selected to be some small positive value.

2 The parameters K_V, I_P, T, C_2 and R_2 are always positive.

3 The expression $K_V I_P T R_2$ is << 1 (This will always be the case for any realistic set of system parameter values).

Thus the only inequality of Y where the expression in equation (5.19) holds is:

$$Y < 2V_0 \tag{5.20}$$

This solution can be graphically illustrated by plotting (5.19) for a range of Y values and a V_0 randomly chosen to be equal to 6. Using equation (5.20) it is expected that in this case equation (5.19) will only be positive when Y is greater than $2V_0$, or 12. This is found to be the case, see Figure 5.7.

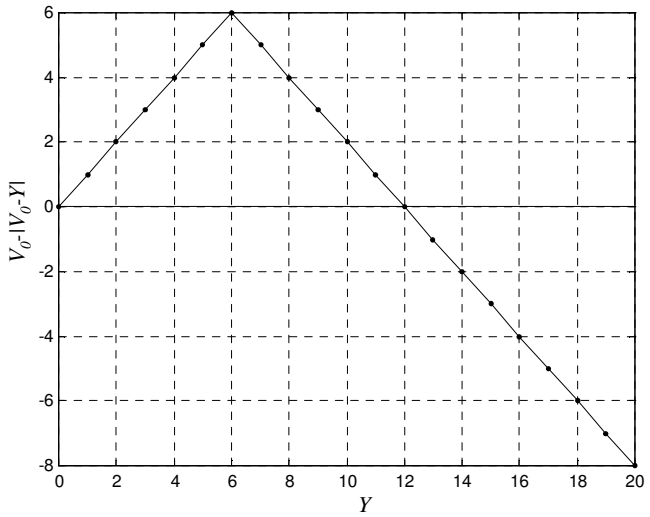

Figure 5.7 Plot of Equation (5.19) for $V_0 = 6$ and a Range of Y

Substituting the double summation expression of Equation (5.18) into Y of (5.20), then the DPLL system is stable when:

$$\frac{K_V I_P T^2}{C} \sum_{i=0}^{m-1} \sum_{k=0}^{i-1} \left(1 - K_V I_P RT\right)^{i-k-1} V_C(k) < 2V_0 \tag{5.21}$$

To prove that the stability criteria in equation (5.21) is independent of the initial control voltage V_0, consider the first four iterations of the left hand side of equation (5.21) from zero up to three, where $A = K_V I_P T^2/C$, and $B = 1\text{-}K_V I_P RT$, as in Table 5.4.

Iteration	Control Voltage
0	$V_0=V_0$
1	$V_1 = V_0$
2	$V_2 = V_0+AV_0$
3	$V_3= V_0+A(V_0(B+1)+V_1)=V_0(1+A(B+1)+A)$

Table 5.4 First Four Iterations of Control Voltage

As illustrated by the first few iterations in Table 5.4, it can be seen that all samples of the control voltage (including samples from $4 \rightarrow \infty$) comprise of previous V_m. In fact any V_m can be reduced to V_0 times a set of A and B equations. The left hand side (LHS) of equation (5.21) can be simplified to the expression:

$$LHS = V_0(1 + \cdots + f\left(A, B\right)) = V_0 X \tag{5.22}$$

where $f(A,b)$ is some function containing only A and B terms. Substituting the result of equation (5.22) back into the LHS of equation (5.21) it can be deduced that the stability criteria is:

$$V_0 X < 2V_0 \tag{5.23}$$

Therefore the boundary is X=2 and is independent of V_0, the initial control voltage offset.

5.3.3 Calculation of Number of State Space Points for One Half Segment of the System Trajectory.

In Section 5.3.1 it was found that the value of the control voltage at any sample point, V_n, could be estimated using equation (5.15). In order to find the control voltage at the phase error zero crossing, V_m (see Figure 5.3), it is necessary to determine the number of periods, m, required to reach the zero crossing. The simplest and most effective way of estimating m is with the linear approximation model[ix]. To estimate m the linear error transfer function given in equation (5.24) is utilised.

$$H_e(s) = \frac{C_2 s^2}{C_2 s^2 + K_V I_P R_2 C_2 s + K_V I_P} \tag{5.24}$$

To determine the phase error zero crossings the response of the DPLL system in the time domain, the inverse Laplace transform of frequency step response of the phase error transfer function is calculated, as shown in (5.25). To estimate the zero crossings, the phase response is equated to zero and solved.

$$L^{-1}\left(\frac{2\pi\Delta_F}{s^2} H_e(s)\right) = 0 \tag{5.25}$$

where Δ_F is the frequency step size. Solving (5.25) gives equation (5.26):

$$g(t)\sin\left(\frac{t}{2}\sqrt{\frac{K_V I_P\left(4 - K_V I_P R^2 C\right)}{C}}\right) = 0 \tag{5.26}$$

where $g(t)$ is a function of time. As the left hand side of equation (5.26) is of the form $g(t)Sin(x(t))$, which is zero when $x(t)=0,\pi,2\pi,3\pi,\ldots$, then the first zero crossing

---

[ix] It is worth noting at this point that using the linear model to determine m in what has heretofore been a piecewise linear analysis does not reduce the accuracy of this technique. This is because we only require an approximate value of m and then round it up to the next integer.

after $t = 0$ occurs when $x(t)=\pi$. Solving this gives equation (5.27), the time of the first zero crossing.

$$t_m = \frac{2\pi}{\sqrt{(K_V I_P (4 - K_V I_P R_2^2 C_2))/C_2}} \tag{5.27}$$

It is possible to estimate the number of samples m, (the solid arc of the system trajectory in Figure 5.3), by considering that if t_m is the time taken to reach the first zero crossing and T, the period, is the time of one sample then the number of periods to reach t_m is equal to t_m/T which is equal to $t_m F_R$. This solution however is not an integer, thus it is required to round $t_m F_R$ up. Finally the solution may fall either side of the phase error zero crossing, so to ensure that the state space sample point steps over the zero crossing (this will benefit the analysis at a later stage) an extra 1 is added to the number of samples. The final solution to m is given in equation (5.28) below.

$$m = \lceil t_m F_R + 1 \rceil \tag{5.28}$$

Finally, substituting the solution to equation (5.27) into (5.28) gives the solution to the number of samples required to reach the phase error zero crossing.

$$m = \left\lceil \frac{2\pi F_R}{\sqrt{(K_V I_P (4 - K_V I_P R_2^2 C_2))/C_2}} + 1 \right\rceil \tag{5.29}$$

Thus the value of the control voltage at the zero crossing of the phase error, V_m, can be determined by using equation (5.29) and solving equation (5.15) for $n = m$.

Earlier in this section it was discussed that the accuracy of this piecewise linear method depended on an accurate estimate of V_m (and thus the value of m). If m is too small (less than 4) then the trajectory will be inaccurate. In equation (5.29) F_R is the largest parameter thus for high frequency systems this is not an issue. This is illustrated in Figure 5.8 below for a second order PLL system. For a range of reasonable filter time constant values, τ, the number of state space sample points of one half segment of the system trajectory becomes small for reference frequencies of less than 15 MHz.

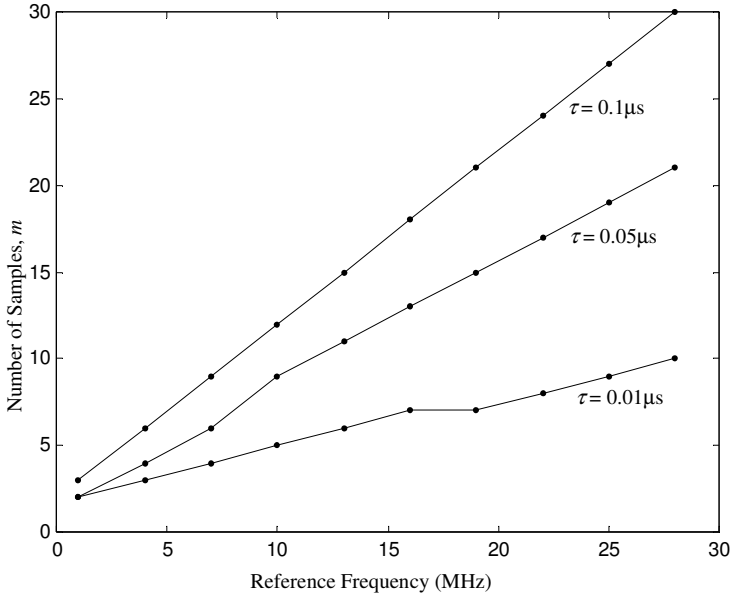

Figure 5.8 Plot of Number of State Space Sample Points for Varying Reference Frequency and Range of Filter Time Constants τ

5.3.4 Closed Form Solution of the Phase Error Zero Crossing Control Voltage

In the previous sections an estimate of the control voltage at the phase error zero crossing was found. This solution is found by evaluating equation (5.15) for m iterations where m is defined by equation (5.29). In this section a closed form solution to the control voltage at the phase error zero crossing (5.15) is presented. This is achieved by rewriting equation (5.15) where $A = -K_V I_P T^2/C_2$ and $B = 1-(K_V R_2 I_P T))$.

$$V_m = V_0 + A\sum_{i=0}^{m-1}\sum_{k=0}^{i-1} B^{i-j-1} V_C(k) \tag{5.30}$$

By looking at the first few iterations of this equation, using the same procedure as was used earlier to break-down the phase error equation in Table 5.2, it can be deduced that this double summation equation can be replaced with equation (5.31). Note that the derivation of equation (5.31) is involved so it is not included here but is derived in Appendix B.

$$V_m = V_0 \begin{pmatrix} 1 + A\Lambda_1 \\ +A^2\Lambda_2 \\ +A^3\Lambda_3 \\ \vdots \\ +A^{\left\lceil \frac{m}{2} \right\rceil}\Lambda_{\left\lceil \frac{m}{2} \right\rceil} \end{pmatrix} \tag{5.31}$$

where Λ_1 up to $\Lambda_{\lceil m/2 \rceil}$[x] are a set of parameters defined as equations (C.1-C.5) in appendix C. Since parameter $|A|$ is always << 1 (since the parameter F_R^2 is a large number and therefore T_2 is small), $|A|$ becomes less significant as the exponent of A is increased. In fact it is found that terms with exponents of A greater than four are insignificant and have negligible influence on the final value of V_m. So equation (5.31) can be simplified to:

$$V_m = V_0 \left(1 + A\Lambda_1 + A^2\Lambda_2 + A^3\Lambda_3 + A^4\Lambda_4\right) \tag{5.32}$$

Equation (5.32) is the closed form solution of the control voltage after m samples.

5.3.5 Stability Boundaries of the Second Order DPLL

In this section the stability boundary of the second order DPLL is considered. Now that the control voltage at the phase error zero crossing, V_m, can be estimated (using either numerical iteration as in Section 5.3.1 or using the closed form solution of the previous subsection) it is possible to determine the stability of the DPLL using V_m and the set of stability criteria given in Table 5.1 earlier. From these results the

---

[x] Note that the partial square brackets $\lceil x \rceil$ is the 'ceiling' of the value x. That is x rounded up to the nearest integer.

system stability boundary can be plotted. The stability boundary is considered in due course; however first consider the stability performance parameter P_{in}, this percentage pull-in rate, in terms of V_m as given in equation (5.33).

$$P_{in} = 100\frac{V_0 + V_m}{V_0}\% \tag{5.33}$$

If the pull-in percentage is negative, the system is unstable otherwise the system is stable. Combining equations (5.32) and (5.33) the stability criterion can be rewritten as:

$$P_{in} = 2 + A\Lambda_1 + A^2\Lambda_2 + A^3\Lambda_3 + A^4\Lambda_4 > 0 \tag{5.34}$$

Note that this is independent of the initial VCO control voltage offset V_0. As expected from the results of Section 5.3.3; the initial condition has no affect on the stability boundary.

The stability boundary can be drawn by equating equation (5.34) to zero and can then be compared to the traditional linear stability boundary. It is also possible to tailor the stability boundary to any desired system requirements by choosing a P_{in} value of greater than zero, and therefore it is possible to find a stability boundary with a faster pull in rate. This is illustrated in Figure 5.9, the stability boundaries of the proposed second order technique are determined for a pull-in rate of 1%, 20% and 40% and are shown along with linear PLL boundary [27] and a stability boundary defined by a number of circuit level simulations.

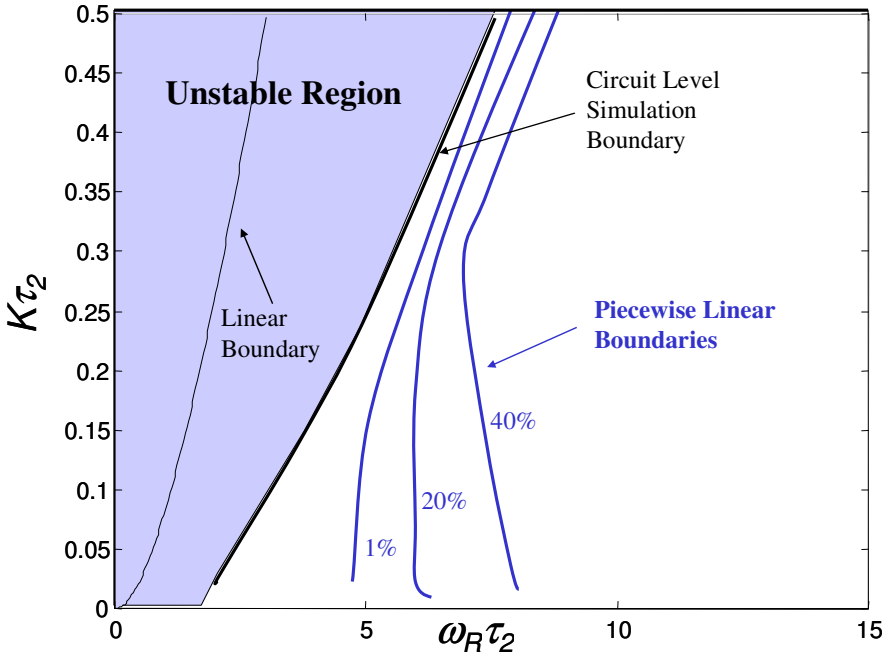

Figure 5.9 Stability Boundaries of 1 GHz Second Order DPLL According to the Linear and the Proposed Method

It can be seen in Figure 5.9 that the circuit level model of the DPLL system suggests that Gardner's prediction is not conservative enough and does not guarantee stability, as we expect from the results of Section 3.3.2 and is suggested by Van Paemel [3]. In fact there is a significant stable region defined by Gardner, where the circuit level model predicts instability. This discrepancy is the reason why DPLL designers need to complement linear design methods with rule of thumb, circuit level simulations and empirical design.

The circuit level boundary also verifies that the proposed piecewise linear technique is more accurate, producing more conservative results than the linear boundary which are inside the stability region of the DPLL system. The conservatism of the piecewise linear method can be verified by considering that the piecewise linear boundaries of Figure 5.9 were determined by utilising the charge approximation. The error introduced by this approximation was explained in section 4.2.2 and is given in equation (4.28). It resulted in the approximated control voltage being slightly greater than the actual control voltage. If this is considered in terms of the percentage pull-in rate given in equation (5.33), assuming that V_m is the actual control voltage and that it is negative, then the approximated percentage pull-in rate is:

$$P_{in\_approx} = 100\frac{V_0 - (V_m + V_{error})}{V_0} \tag{5.35}$$

$$\Rightarrow P_{in\_approx} = 100\frac{V_0 - V_m}{V_0} - 100\frac{V_{error}}{V_0} \tag{5.36}$$

$$\Rightarrow P_{in\_approx} = P_{in} - 100\frac{V_{error}}{V_0} \tag{5.37}$$

From equation (5.37) it can be seen that the approximated pull-in rate is equal to the actual pull-in rate minus some small error. So, for example, if a marginally stable system is considered (P_{in} = 0%) then the approximated pull-in rate will be some negative value, i.e. the piecewise linear prediction is unstable. Likewise if the actual pull-in rate of a particular system is 20%, then the approximated value will be something less than this. Thus the charge approximation has a stabilising effect on the model keeping the predicted stability boundary on the conservative side of the actual DPLL stability boundary. Any DPLL system designed with parameters chosen from within this predicted boundary will be stable. This is a significant advantage over the linear boundary and further illustrates the inaccuracies of applying the linear model to the DPLL system.

While comparing the proposed technique to additional stability methods such as [13, 26, 43] is desirable, it is not possible due to the nature of these methods, and the difficulty in plotting any global stability boundary with such methods. This is due to the dependency that exists between the defined parameters, i.e. changing one parameter has an effect on others, thus a point on the stability plane of Figure 5.9 can not be easily selected, as is the case with the circuit level simulation.

Finally consider the following design example, where the proposed piecewise linear stability criterion is used to define the stability boundary and pull-in rates of a 100 MHz second order DPLL. The predicted stability plane for this system is plotted in Figure 5.10, where the three axes are $K\tau_2$ and $\omega_R\tau_2$, similar to Figure 5.9, and the third axis is the percentage pull-in rate.

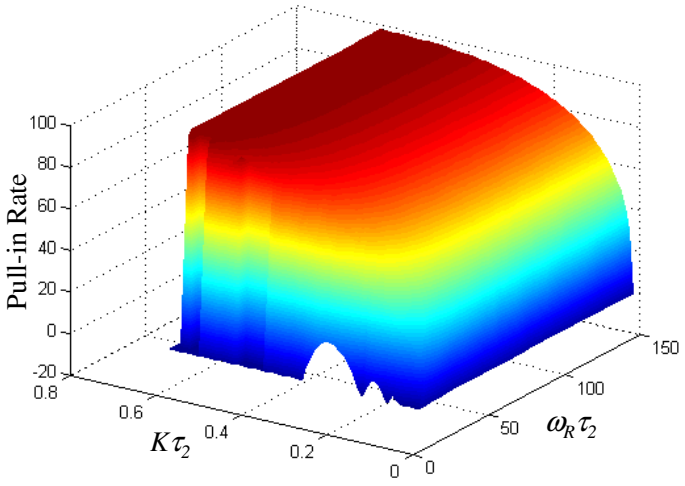

Figure 5.10 Stability Boundary for 100 MHz system

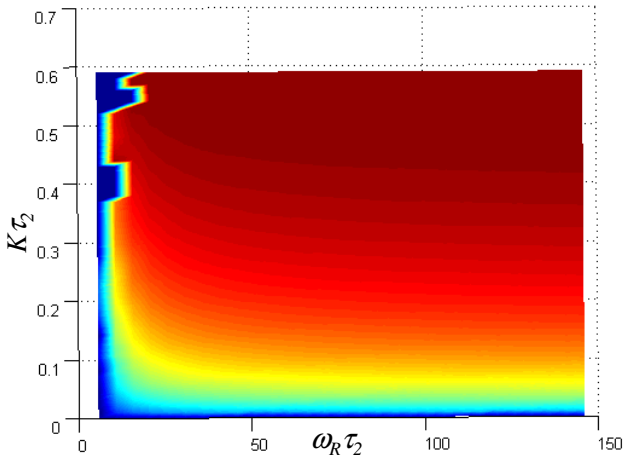

Figure 5.11 Plan View of Figure 5.9

Figure 5.11 is a plan view of Figure 5.9. The bluer the region the less stable the DPLL is. The best choice of $K\tau_2$ and $\omega_R\tau_2$ for a stable system are, 0.25 and 150 respectively. Using these values we can simulate the response of a stable 100 MHz CP-PLL with rounded component values of K_V = 66 Mrad/s/V; I_P = 10x10^{-6}A; R_2 = 10 kΩ; C_2 = 24pF; Figure 5.12 shows that as expected the DPLL response is stable. The voltage jumps inherent to the second order DPLL can also be seen in Figure 5.12 these have an expected value of $I_P R_2$.

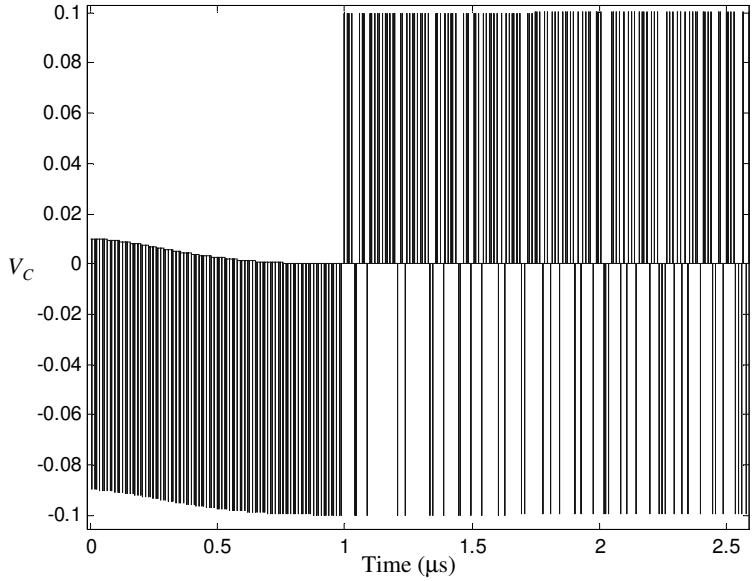

Figure 5.12 Transient Response of the 2nd Order System

5.3.6 Calculation of the Control Voltage on the Phase Error Zero Crossing

The control voltage V_m has already been calculated, however it has already been stated that this value of V_m at sample point m is unlikely to fall on the phase error zero crossing. In other words V_m will not correspond with V_x, as shown in Figure 5.13, because the last sample m will not fall exactly on the phase error zero crossing, but will cross that line by some value d. In this section a linear interpolation methodology is used to calculate the value of the control voltage on the zero crossing from the knowledge of the last two sample values.

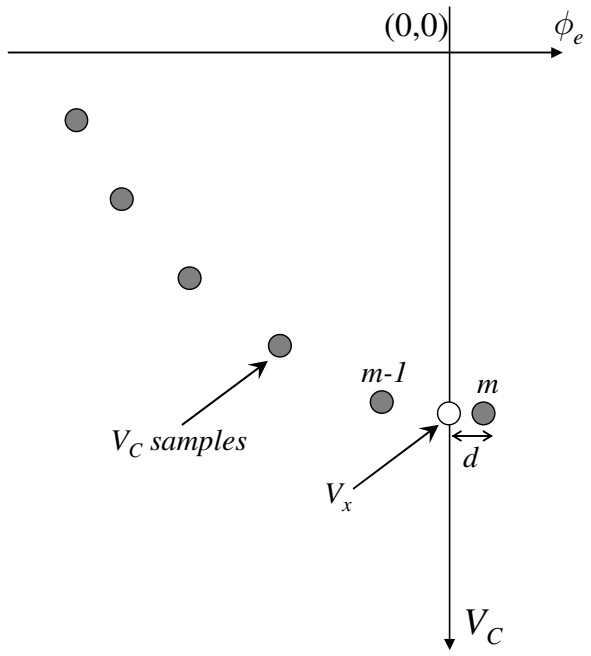

Figure 5.13 State Space Samples

The estimation of V_x is achieved by utilising equation (5.15) and solving for *m* samples, i.e. If the control voltage values at sample points *m-1* and *m* are both known then it is possible to calculate the value of V_x by using a linear interpolation, see equation (5.38).

$$V_x = V_{m-1} - \phi_{m-1} \frac{|V_m| - |V_{m-1}|}{|\phi_m| + |\phi_{m-1}|} \tag{5.38}$$

where (V_m, ϕ_m) and (V_{m-1}, ϕ_{m-1}) are the co-ordinates of the samples *m* and *m-1* respectively in Figure 5.13. However despite the obvious benefits of using equation (5.38) it is not used in this model as the additional model complexity exceeds the benefit of reduced error. The existing error is minimal and reduces as the reference frequency is increased.

5.3.7 Estimation of System Settling Time

In this section an estimate of t_{LCK}, the system lock time or settling time is determined. This is achieved by utilising the estimate of t_m, the time of one half cycle of the system trajectory, as derived earlier in equation (5.27) and from the system pull-in rate given in equation (5.33).

To determine the lock time, a new parameter κ is introduced, this is the number of half cycles of the state space trajectory in Figure 5.3, required to reach lock. Thus the system lock time can be estimated as:

$$t_{LCK} = \frac{\kappa}{\omega_x} \tag{5.39}$$

where ω_x is the natural frequency of the system, and ω_x is equal to $1/t_m$ where t_m is as given in equation (5.27). Using the definition of stability of a DPLL given at the start of this chapter, t_{LCK} can be defined as the time it takes the system to settle to less than some value of V_{LCK} away from the equilibrium (or until V_C settles within the predefined V_{LCK} boundary). Using this definition of stability the system is considered locked when:

$$V_{LCK} < V_0 \left(1 - \frac{P_{in}}{100}\right)^m \tag{5.40}$$

where m is as defined earlier (the number of state space sample points in the left half segment of the state space trajectory). The exponent m is due to the fact that the control voltage starts at an offset of V_0 and is pulled in by a factor of $P_{in}/100$ every sample m[xi]. Solving equation (5.40) for m gives:

$$m > \frac{\log_{10}\left(\frac{V_{LCK}}{V_0}\right)}{\log_{10}\left(1 - \frac{P_{in}}{100}\right)} \tag{5.41}$$

Since m is a calculation of the number of half cycles required to reach lock, κ is equal to m:

[xi] Note that because equation (5.36) assumes that the system is pulling towards the equilibrium, and not away from it, this solution is only valid when the system is stable.

$$\kappa = \frac{\log_{10} \frac{V_{LCK}}{V_0}}{\log_{10}(1 - \frac{P_{in}}{100})} \tag{5.42}$$

From equations (5.27, 5.39, and 5.42) the lock time t_{LCK} can be estimated as:

$$t_{LCK} = \frac{t_m Log_{10} \frac{V_{LCK}}{V_0}}{Log_{10}(1 - P_{in})} \tag{5.43}$$

Finally consider the design example that was given at the end of Section 5.3.5, using equation (5.43) the lock time is estimated to be approximately 1.6μs, where V_{LCK} is chosen to be $\pm 1 \times 10^{-4}$V. Looking at the circuit level response of the DPLL system in Figure 5.12, and ignoring the inherent RC jumps, a lock time of 1.6μs is a reasonable estimate.

5.4 High Order Stability Boundaries

In the previous section a second order stability boundary solution was proposed using a piecewise linear method, this was a closed form solution allowing for the calculation of the boundary without the need for a large number of iterations. This stability methodology can also be used to model higher orders. As mentioned previously the charge approximated behavioural equations are beneficial because they remove the high order differential terms from the filter equations, making the solution of higher order equations mathematically feasible. In this section the high order piecewise linear model is derived and used to determine the stability boundary of high order systems, something that is not possible with existing behavioural models.

Similar to the second order piecewise linear model equations derived in the previous section, the third order stability boundaries can be derived by starting with the charge approximated second order filter behavioural equations, given in equations (4.19 – 4.23), and making the following observations:

1 In the case of higher order systems the model is solved using a piecewise linear solution. In the second order case sampling the model every T seconds

was sufficiently accurate; this is not the case with higher order systems. Considering that the loop filter is now of second order, the straight line solution to the integral of V_C is not accurate. However this is overcome by iterating the system behavioural equations between samples in a piecewise linear fashion, as shown in Figure 5.14.

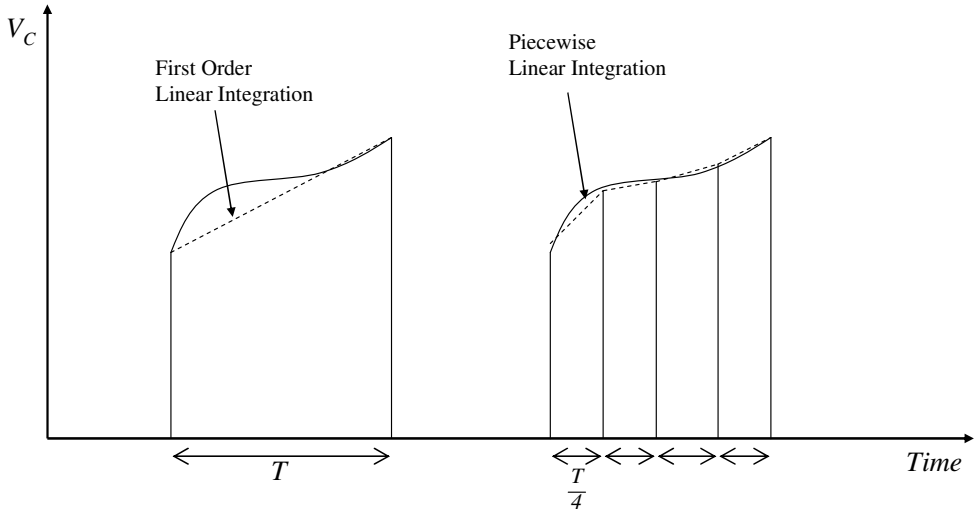

Figure 5.14 Piecewise Linear Integration

The model is now sampled every T' seconds, where T' is the sample time T divided by g, and g is the number of piecewise linear iterations. The choice of g will define the accuracy of the integration; however this needs to be traded with the expected increase in simulation time. Note that from this point forward it is assumed that T is the time period of one piecewise linear iteration. i.e. $T = T'$.

2 With higher order systems the state transitions need to be considered. This is because each capacitor in the loop will charge up during the boost period T_B and discharge during the coast period T_C. Therefore the current into the loop filter will be $-I_P$ for time period T_B and 0 for the remainder of the period T_C, where T_B is given in equation (5.6), and $T_C = T\text{-}T_B$. Therefore the integral of V_C depends on which time period is being iterated (T_B or T_C). The integral of V_C can be calculated as given in equation (5.44).

$$\int V_C dt = T_P V_C(t_k) + \frac{T_P^2 I_3(t_k)}{2C_3} \tag{5.44}$$

where $I_3(t_k)$ is calculated using equation (4.21), and T_P is either T_B or T_C depending on the loop state, as defined earlier.

The third order phase error signal is calculated using equation (5.45).

$$\phi_e(t_{k+1}) = \phi_e(t_k) - 2\pi K_V \int V_C dt \tag{5.45}$$

3 In the case of the third order DPLL the control voltage V_C is the voltage across capacitor C_3, as in Figure 2.19. V_C is given in equation (5.46).

$$V_C(t_{k+1}) = \frac{Q_3(t_{k+1})}{C_3} \tag{5.46}$$

where $Q_3(t_{k+1})$ is calculated using equation (4.22).

Using the methodology described above the third order state trajectory can be estimated by iterating the determined behavioural equations and plotting the system state variables against each other. Note that the second order DPLL has two state variables; these are the phase error ϕ_e and the control voltage V_C. However high order DPLL systems have additional state variables that need to be considered, specifically the charge on each additional filter capacitor. However if V_C reaches a stable equilibrium point then the filter capacitor state variables have also reached an equilibrium point. Therefore when considering the stability of a high order system, we need only monitor ϕ_e and V_C. Unlike the second order DPLL, the relevant high order DPLL stability boundary equations cannot be solved into a closed form solution as it is not mathematically tractable. However the relevant equations can be easily solved using numerical iteration of the model equations given in Appendix D. Thus the first zero crossing of the control voltage V_m for the third order system can be estimated as in Figure 5.3 by using equations (5.44 – 5.46), equations (4.19 – 4.23) and numerical iteration.

From the solution to V_m the stability boundary is found for the third order DPLL and these boundaries can now be compared to the traditional linear boundaries that were

plotted in Section 3.2.2 earlier. In Figure 5.15, the stability boundaries for the third order system are plotted for b =8, where $b = 1+C_2/C_3$. This definition of b is originally given in [27]. Similarly the third order boundary is plotted for a value of b =16.

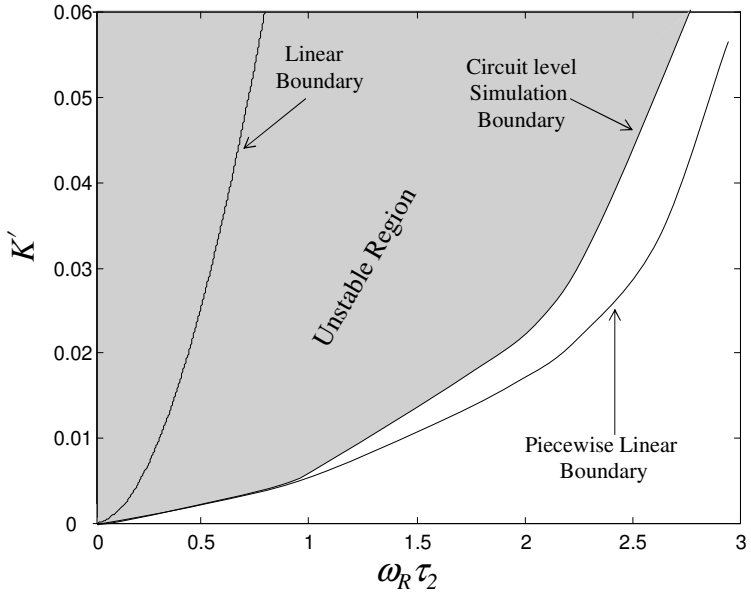

Figure 5.15 Third Order Stability Boundaries for b = 8

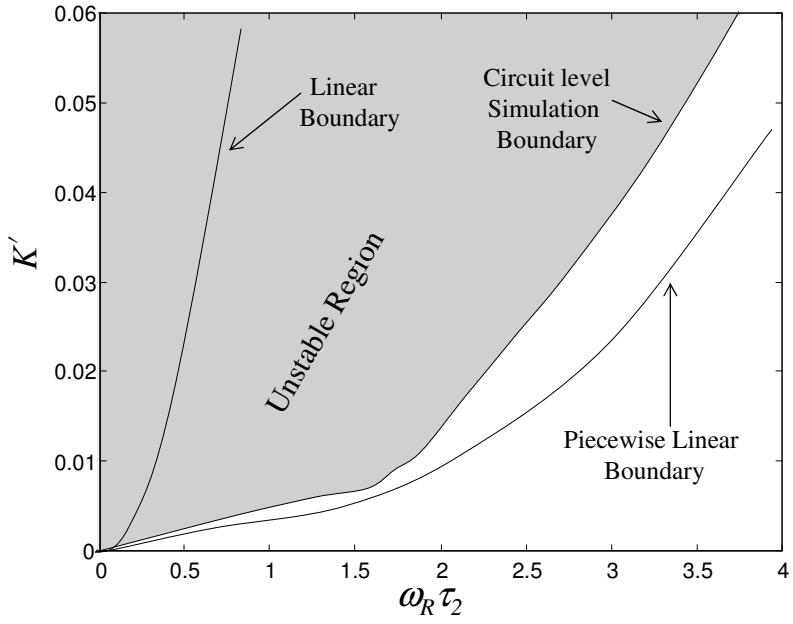

Figure 5.16 Third Order Stability Boundaries for b = 16

Similar to the second order case, both Figure 5.15 and Figure 5.16 illustrate the weakness of the linear approximated boundaries. To further illustrate this consider an example using the proposed technique to define the stability boundary of a third order DPLL with a 1 GHz reference signal, a divide ratio of 1 and a value of *b* equal to 8. The predicted boundary is plotted in Figure 5.17, along with a linear stability boundary.

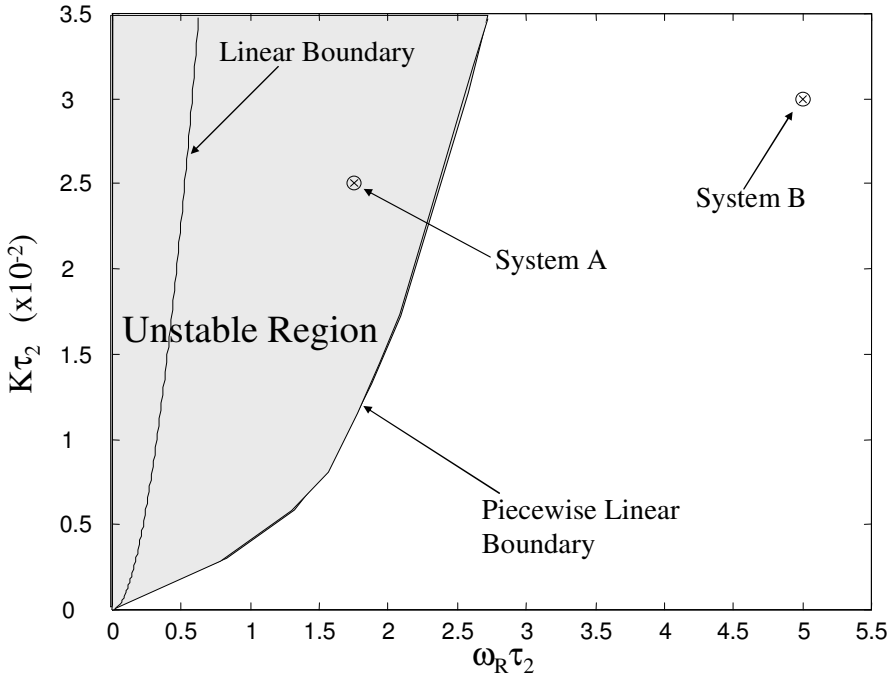

Figure 5.17 Predicted Stability Boundary for 1 GHz System

The shaded section in Figure 5.17 is the unstable region of the DPLL as defined by the proposed methodology. Using this prediction of the boundary, two systems are considered, system A and B. The location of system A in Figure 5.17 (where $K\tau_2$ and $\omega_R\tau_2$ are equal to 0.025 and 1.75 respectively) will define its gains and loop filter parameter values. Note that system A lies in the region between the predicted piecewise linear boundary and the linear prediction, therefore it is being predicted as unstable by the piecewise linear method but stable by the traditional linear method. The response of system A is determined using a circuit level simulation and is plotted in Figure 5.18.

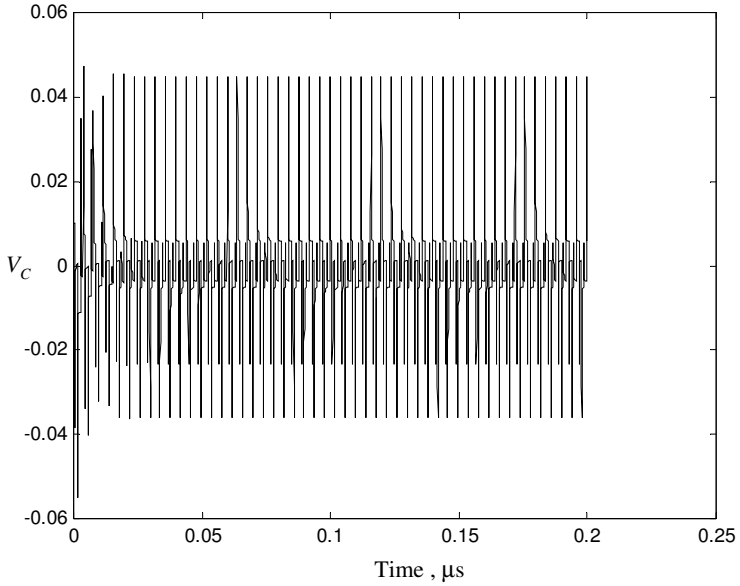

Figure 5.18 Response of System A

We can see that as predicted by the piecewise linear method the response of system A is unstable, this is counter to the stable prediction of the linear method. In contrast System B is well within the stable region of both predictions with a choice of $K\tau_2$ and $\omega_R\tau_2$ of 0.03 and 5 respectively and should therefore be stable. The response of system B is again determined using a circuit level simulation and is plotted in Figure 5.19 below.

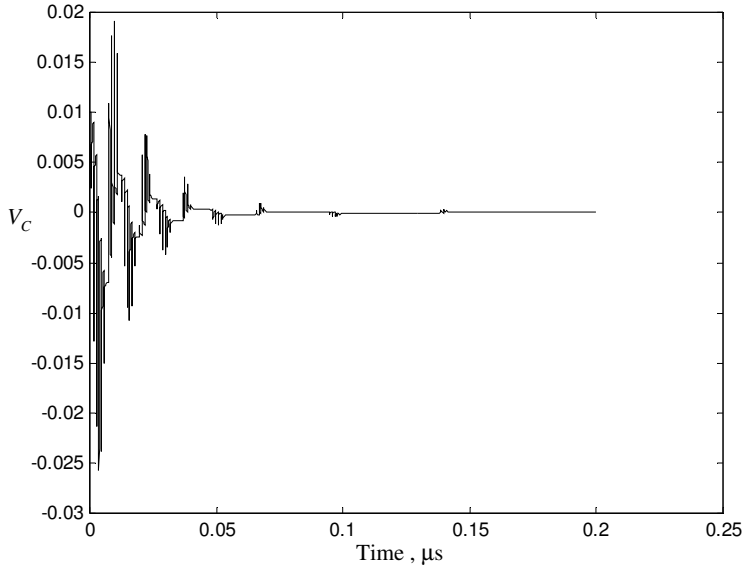

Figure 5.19 Response of System B

In the case of system B the system is found to be stable (Figure 5.19). These results show that, similar to the second order system, the third order piecewise linear stability boundary technique provides a much better prediction of the stability boundary than linear methods.

In this chapter second and third order stability boundaries were estimated and compared with existing stability boundary. It is also possible to increment this stability boundary to higher orders. Fourth and fifth order piecewise linear stability equations are given in Appendix D, unlike the case of the second and third order systems there are no references to compare the results with. The fourth order system boundary can be determined using the same methodology as the third order earlier. However as the system order is increased there are an increasing number of system parameters that need to be considered, thus it is not possible to plot the boundaries without defining some relationship between some of these parameters and thus reducing the number of variables to a manageable amount. The solution to the increasing number of design parameters is to describe the loop filter performance using just one parameter, the filter cut-off frequency. This can be achieved by using filter prototypes to place the filter poles, thus the whole DPLL system can be defined by only three parameters regardless of the loop order; these are the two gain components K_V and K_P and the filter prototype cut-off frequency ω_c, this design methodology is considered in detail in Chapter 6.

5.5 Conclusion

The piecewise linear and charge approximation methodologies that are proposed in this thesis are both used in this chapter to accurately determine the stability boundaries of arbitrary order DPLL systems. This methodology offers advantages over circuit level simulation and existing linear DPLL models in terms of simulation time and accuracy respectively. The other advantages of this methodology lie in the fact that the stability boundary determined for a particular order of system is independent of any initial conditions, and unlike the existing linear boundaries it accurately models the inherent DPLL nonlinearities. The fact the stability boundaries are found to be independent of initial conditions, proves that if any

trajectory is determined for a particular system then this will accurately represent the response of the system for any set of initial conditions or subsequent path through state space.

The stability design methodology proposed in this section is utilised to derive a closed form stability boundary of the second order DPLL system. Though this solution is expansive, it is mathematically tractable and is found to better define the stability region of the second order DPLL than existing methods. The stability boundaries that are determined using the piecewise linear method proposed in this section are a significant improvement on the linear equivalent boundaries. Such boundaries are still used today as a start point in the design of stable DPLL systems. The improvements that are achieved using the piecewise linear boundary will make the current design process significantly faster, more efficient and reliable.

Another advantage of the proposed piecewise linear method is that it can be extended to higher orders. This means that accurate high order stability boundaries can be determined and used to design stable high order DPLL systems. Using this method a greater understanding of high order DPLL stability and performance can be determined.

CHAPTER 6
DESIGN OF HIGH ORDER DIGITAL PHASE-LOCK LOOPS

6.1 Introduction

The most common DPLL systems designed today have loop orders of two and three. Higher loop orders are considered unreliable due to the increased complexity of model equations; loop stability issues; inaccuracy with the linear model; and lack of a viable and accurate analysis technique. In the preceding chapters the DPLL model complexity and inaccuracy were overcome by utilising a charge approximation of the loop filter equations and a piecewise linear modelling methodology. As a result it is possible to model the DPLL loop with increased accuracy over existing linear methods and without the low loop order restriction that exists with existing methods.

In this section the design of high order DPLL systems using the proposed charge approximated piecewise linear model is considered. This is not a simple task due to the large number of loop parameters, and therefore the large number of possible design combinations. For example consider a fifth order DPLL, to design this loop two gain components (K_V and I_P), and seven loop filter parameters (three resistors and four capacitors) need to be selected. Without any knowledge of the fifth order DPLL stability it would be an arduous task to select stable loop parameters. Utilizing the methodologies proposed in the preceding chapters it is possible to determine stable regions for this fifth order system. However, in the case of the fifth order DPLL, this would involve plotting all nine parameters against each other. Ideally the designer requires a two dimensional plot of the stability boundaries (as in the example of Figure 3.17). To achieve this the number of design parameters needs

to be minimised to a manageable level. One solution is to relate variables to each other, for example let one resistor be some fraction of another. This is desirable to some degree because there are a number of rules-of-thumb for the relationship between different parameter values (pole locations) to ensure stability[xii]. While this approach has the desired effect of reducing the number of design parameters it also has the undesirable effect of tying loop poles together. As the poles are rigidly related, this has the effect of significantly reducing the flexibility and scope of the design process. This design method depends on rule-of-thumb knowledge, which is not ideal. In this chapter an alternative methodology is presented which places all system poles simultaneously by utilising existing filter prototypes; such as Bessel, Butterworth, and Chebyshev. Using filter prototypes means that the loop filter performance can be defined by the filter prototype cut-off frequency, ω_p, thus having the desired effect of reducing the number of design parameters to two (ω_p and the loop gain $K_V I_P$), regardless of the system order.

This design methodology presented in this section equates the DPLL loop system transfer function with the relevant order transfer function of the filter prototype. This results in a set of equations which define the loop filter parameters in terms of the normalised filter prototype coefficients. Thus the filter performance can be described by the prototype cut-off frequency ω_p.

It will also be demonstrated in this section that by using the above approach the system transfer function (and therefore the overall loop performance) is independent of the loop gain $K_V I_P$. This is an important finding as it means that the only design parameter that is critical to the loop performance is ω_p. Finally by combining this filter prototype methodology with the piecewise linear method of the previous

---

[xii] There are many rules-of-thumb for the optimum choice of loop filter parameter values, these were discussed in chapter 2.4.

chapter, it is possible to accurately determine the system lock time for a range of ω_p. From this stability boundary the design parameter ω_p is chosen and then the DPLL loop parameter values can be calculated. This design technique is beneficial to high order systems for a number of reasons:

1. High order filter prototype parameters are known and thus can easily be equated to the relevant order of DPLL;

2. Regardless of the DPLL system order the loop filter performance can be described by one parameter, the filter prototype cut-off frequency of ω_p.

3. The system analysis is simplified, by the need to only consider a limited number of parameters regardless of the system order.

The inaccuracies of applying linear methods to the DPLL have already been discussed in Chapter 3 of this thesis; these are the inaccuracies of the linear approximation. However the design methodology presented in this chapter does use some linear approaches, namely it equates the linear PLL transfer function with the linear filter prototype. This linear section is used solely to reduce the number of design parameters that the designer needs to consider, see Figure 6.1.

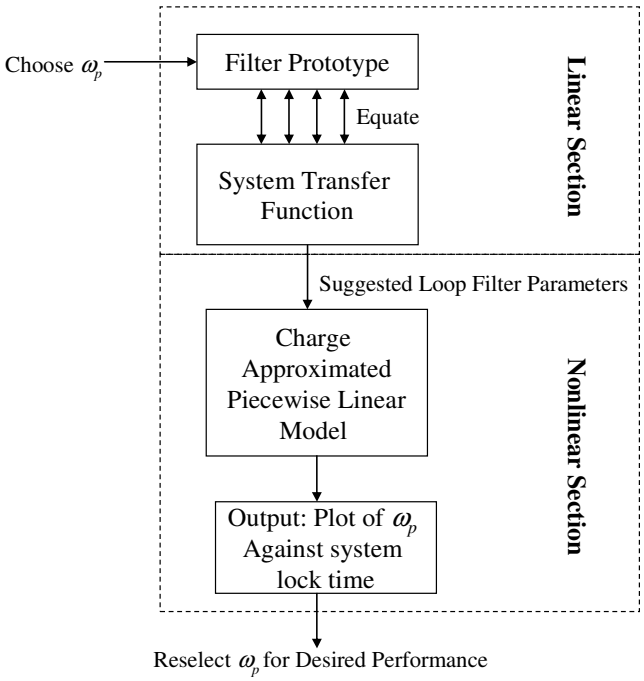

Figure 6.1 Overall Design Procedure

The loop parameter values resulting from the linear section are then used by the nonlinear section as a starting point from which the overall system stability boundary is determined using nonlinear means. In other words, the linear section of this design methodology only selects ω_p for a desired system performance; the nonlinear section determines the actual system performance of the DPLL for parameter values determined by that particular prototype.

As a final design consideration Section 6.3 considers the robustness of the proposed high order design methodology. The design methodology presented in this chapter determines system parameters and performance without any noise considerations, however in any real DPLL system, as discussed in Section 2.5; there are many sources of noise. Thus, it is necessary to consider the effects that these noise sources have on the system performance. While a full noise analysis of the DPLL system is beyond the scope of this thesis, Section 6.3 considers the effects on the system performance due to noise, or slight variations of the linear parameter values. The robustness of the DPLL system will determine how tolerant the system is to such variations.

6.2 High Order DPLL Design Using Filter Prototypes

The choice of loop filter cut-off frequency is the most crucial choice the designer makes in terms of stability and system performance when designing the DPLL. An incorrect choice of ω_c and the system may operate with too much noise, causing instability or bad performance, or the system may over-attenuate the loop signal causing slow lock time and instability. The traditional design techniques outlined in Section 3.2 earlier, do not put forward any explicit recommendation for the choice of ω_c, suggesting only that ω_c should be chosen to be no greater than $1/10^{th}$ of the reference signal frequency ω_R for any system regardless of design specification or filter order. It is left to the designer to use rule of thumb and/or empirical design to make the critical choice of ω_c. In this section, the optimum value of ω_c is determined using filter prototypes, i.e. the linear PLL transfer function is equated with a filter prototype transfer function (effectively letting ω_c equal to the prototype cut-off frequency, ω_p), from this the DPLL parameter values are determined. Finally, by

utilising the piecewise linear methodology of the previous section, the performance of the DPLL, with that particular choice of ω_c, is determined. By plotting the stability plane for a range of ω_c, the optimum system parameters can be selected.

In Section 3.2 earlier the third order DPLL design methodology proposed by Mirabbasi [26] was discussed. This design methodology uses the Bessel filter prototype to optimally place the system poles, and thus optimally design the loop performance. This is a completely linear process which includes the inaccuracies of the linear approximation, however because Mirabbasi only considers the third order loop and uses rule-of-thumb to chose the additional ripple capacitor ($C_3 = C_2/5$), the linear approach in this case is accurate. Also because of the low order of the system and the rule-of-thumb choice of C_3, the prototype is not applied as effectively as it could be. Mirabbasi specifically chooses the Bessel class of filter prototype due to its property of a linear phase offset in the filter pass-band. This property, however, is an unnecessary requirement as the DPLL reference frequency is constant, and any phase offset is corrected in the normal operation of the DPLL. This can be verified by comparing the phase offset of the traditional DPLL to filter prototype phase offsets in the pass-band, as in illustrated in Figure 6.2 below for a cut-off frequency of 10 MHz.

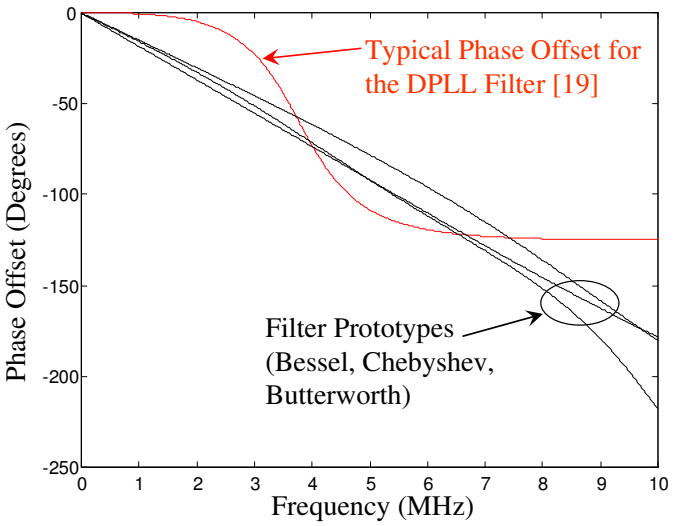

Figure 6.2 Phase Offset for Traditional DPLL and Filter Prototypes

The traditional DPLL phase offset is considerably less linear in the pass-band than any of the selected prototypes. For this reason, the design method need not be restricted to the Bessel prototype but may also include frequency selective prototypes, such as Butterworth and Chebyshev. This is considered in detail in the following section.

6.2.1 PLL Pole Placement using Filter Prototypes

The DPLL filter prototype design methodology determines component values of the filter by equating the filter prototype denominator with the denominator of the PLL loop transfer function *H(s)* as in equation (6.1).

$$H(s) = \frac{K_P K_V N F(s)}{Ns + K_P K_V F(s)} \tag{6.1}$$

where *F(s)* is the relevant loop filter transfer function as described in Section 2.4.1 of this thesis.

The fourth order DPLL transfer function can be determined by substituting the relevant loop filter transfer function from Table 2.2 into equation 6.1:

$$H(s) = \frac{K_P K_V N C_2 R_2 s + K_P K_V N}{\left(\begin{array}{l} N C_2 C_3 C_4 R_2 R_4 s^4 + N\left(C_2 C_3 R_2 + C_2 C_4 R_4 + C_3 C_4 R_4 + C_2 C_4 R_2\right) s^3 + \\ + N\left(C_2 + C_3 + C_4\right) s^2 + K_P K_V C_2 R_2 s + K_P K_V \end{array} \right)} \tag{6.2}$$

Note that the fourth order DPLL has only one zero[xiii] located at $s = -1/R_2C_2$ and four poles. As the design methodology used here equates the filter prototype and the transfer function denominator only, and not the numerators, the filter prototype is

[xiii] The single zero is due to the structure of the DPLL loop filter and due to the fact that the filter must be transconductance in nature.

required to be an all pole system with no zeros. The classes of filter prototype that satisfy this requirement are the Bessel, Butterworth and Chebyshev (type 1) filter prototypes. Each prototype needs to be considered individually to determine the most effective prototype in terms of placing the DPLL system poles; this is considered later in Section 6.2.4. However for the moment, a generic fourth order filter prototype transfer function is used, as given in equation (6.3) below, where the parameters α, β, δ, ε and χ are the normalised coefficients, given in Table 6.1, and ω_c is the DPLL bandwidth.

$$H_{\mathrm{Pr}ototype}(s) = \frac{\chi\omega_C^4}{s^4 + \alpha\omega_C s^3 + \beta\omega_C^2 s^2 + \delta\omega_C^3 s + \varepsilon\omega_C^4} \tag{6.3}$$

Prototype	**Bessel**	**Butterworth**	**Chebychev (*R*=0.1)**	**Chebychev (*R*=0.5)**	**Chebychev (*R*=0.969)**
α	3.124	2.6131	1.804	1.197	0.9637
β	4.392	3.4142	2.627	1.717	1.464
δ	3.201	2.6131	2.026	1.025	0.7541
ε	1	1	0.8285	0.3791	0.2795
χ	1	1	0.819	0.3578	0.25

Table 6.1 Normalised Filter Prototype Coefficients

Equating the denominator of equation (6.2) with that of equation (6.3) results in the following set of expressions given in equations (6.4 – 6.7).

$$\alpha\omega_C = \frac{C_2C_3R_2 + C_2C_4R_4 + C_3C_4R_4 + C_2C_4R_2}{C_2C_3C_4R_2R_4} \tag{6.4}$$

Where K is the system gain K_VK_P.

$$\beta\omega_C^2 = \frac{C_2 + C_3 + C_4}{C_2C_3C_4R_2R_4} \tag{6.5}$$

$$\delta\omega_C^3 = \frac{K}{NC_3C_4R_2} \tag{6.6}$$

$$\varepsilon\omega_C^4 = \frac{K}{NC_2C_3C_4R_2R_4} \tag{6.7}$$

By inspection of equations (6.6) and (6.7) it can be seen that:

$$\varepsilon\omega_C^4 = \left(\frac{K}{NC_3C_4R_4}\right)\frac{1}{R_2C_2} = \frac{1}{R_2C_2}\delta\omega_C^3 \tag{6.8}$$

$$\Rightarrow \varepsilon\omega_C = \frac{\delta}{R_2C_2} \tag{6.9}$$

$$\Rightarrow R_2 = \frac{\delta}{\varepsilon\omega_C C_2} \tag{6.10}$$

Because there are five unknown parameter values and only four simultaneous equations, it is necessary to define two new ratios, M_1 and M_2 as in equations (6.11) and (6.12) below.

$$C_3 = \frac{C_2}{M_1} \tag{6.11}$$

$$R_4 = \frac{R_2}{M_2} \tag{6.12}$$

The effects of the ratios M_1 and M_2, and the optimum choice of these ratios are considered in Section 6.2.3 below.

Using equations (6.4), (6.11), and (6.12), it is possible to determine C_4 in terms of C_2 as follows. Rewrite equation (6.4) as:

$$\alpha\omega_C\left(C_2C_3C_4R_2R_4\right) = C_2C_3R_2 + C_2C_4R_4 + C_3C_4R_4 + C_2C_4R_2 \tag{6.13}$$

From (6.7) $C_2C_3C_4R_2R_4$ is equal to $\frac{K}{N\varepsilon\omega_C^4}$ and (6.13) can be rewritten as:

$$\frac{\alpha K}{N\varepsilon\omega_C^3} = C_2C_3R_2 + C_2C_4R_4 + C_3C_4R_4 + C_2C_4R_2 \tag{6.14}$$

Substituting equations (6.11) and (6.12) into equation (6.14) gives:

$$\frac{\alpha K}{N\varepsilon\omega_C^3} = \frac{C_2^2R_2}{M_1} + \frac{C_2C_4R_2}{M_2} + \frac{C_2C_4R_2}{M_1M_2} + C_2C_4R_2 \tag{6.15}$$

Dividing both sides by R_2 and substituting in equation (6.10) leaves:

$$\frac{\alpha KC_2}{N\delta\omega_C^2} = \frac{C_2^2}{M_1} + \frac{C_2C_4}{M_2} + \frac{C_2C_4}{M_1M_2} + C_2C_4 \tag{6.16}$$

Finally divide both sides by C_2 and solve for C_4.

$$C_4 = \frac{\dfrac{\alpha K}{N\delta\omega_C^2} - \dfrac{C_2}{M_1}}{\left(\dfrac{1}{M_2} + \dfrac{1}{M_1M_2} + 1\right)} \tag{6.17}$$

All the loop filter parameter values have been determined in terms of C_2 (except C_2), see equations (6.10 – 6.12) and equation (6.17). The final task is to determine an expression for C_2 that is independent of the other system parameters. This is derived below by starting with equation (6.5), replacing $C_2C_3C_4R_2R_4$ with $\frac{K}{N\varepsilon\omega_C^4}$ (from equation (6.7)), and subbing in the known values of C_3 and C_4 from equations (6.11) and (6.17):

$$\frac{\beta K}{N\varepsilon\omega_C^2} = C_2 + \frac{C_2}{M_1} + \frac{\dfrac{\alpha K}{N\delta\omega_C^2} - \dfrac{C_2}{M_1}}{\left(\dfrac{1}{M_2} + \dfrac{1}{M_1M_2} + 1\right)} \tag{6.18}$$

The above expression has only one unknown parameter, C_2, so solving this expression for C_2 and simplifying leaves:

$$C_2 = \frac{\beta\delta K(M_1M_2+1) - \alpha\varepsilon M_1M_2K}{N\delta\varepsilon\omega_C^2(M_1M_2 + \frac{1}{M_1})} \tag{6.19}$$

Thus, the fourth order DPLL parameter values can be determined from the filter prototype using equations (6.10 – 6.12), (6.17) and equation (6.19). Using this set of parameters, there are four unknown parameters the gain *K*, the filter cut-off ω_c and the parameter ratios, M_1, M_2. This has been reduced down from the original six unknown component values. In the next subsection it will be shown that the DPLL transfer function that is derived by employing the set of filter prototype parameters above, are independent of the system gain *K*, thus reducing the number of design parameters to three.

6.2.2 Proof that the Filter Prototype Derived DPLL is Independent of the System Gain

In this section, it is shown that if the PLL system transfer function is determined using the parameter values derived by the filter prototype values given in equations (6.10 – 6.12), (6.17) and equation (6.19), then the closed loop transfer function is independent of the choice of system gain *K*. First, consider equation (6.19) in terms of *K,* (6.20) shows that C_2 is some scalar product of *K*.

$$C_2 = K\left(\frac{\beta\delta(M_1M_2+1) - \alpha\varepsilon M_1M_2}{N\delta\varepsilon\omega_C^2(M_1M_2 + \frac{1}{M_1})}\right) = aK \tag{6.20}$$

where *a* is some unknown constant value. Also rewriting the other parameter values in terms of *K* and unknowns gives:

$$\Rightarrow R_2 = b\frac{1}{C_2} = \frac{b}{aK} \tag{6.21}$$

$$C_3 = caK \tag{6.22}$$

$$R_4 = \frac{bd}{aK} \tag{6.23}$$

$$C_4 = \frac{eK + fC_2}{g} = \frac{e + fa}{g} K = hK \tag{6.24}$$

where *a, b, c, d, e, f,* and *g* are also unknown constant values. Substituting all these parameter values back into the system transfer function of equation (6.2), and solving *H(s)* in terms of *K* gives:

$$H(s) = \frac{K\left(Nbs + N\right)}{\left(\begin{array}{l} NKbchs^4 + NK\left(bca + bdh + cdbh + bh\right)s^3 \\ +NK\left(a + ca + h\right)s^2 + Kbs + K \end{array}\right)} \tag{6.25}$$

By inspection of equation (6.25) it can be seen that *K* above the line cancels with the *K* values below the line, thus *H(s)* does not depend on the system gain *K*, i.e. any change in *K* is reflected by a proportionate change in the derived system component values.

This is an important finding and is a consequence of the filter prototype derivation methodology given above. It means that in theory the system gain does not affect the system stability and performance, i.e. that as expected, the system bandwidth is set by the filter prototype (and the choice of ω_c) and does not depend on the gain *K*. However, it cannot be forgotten that the choice of K_V and K_P will have an effect on the performance of the VCO and PFD components, in particular in terms of noise suppression and the PFD tracking performance and dc offset. Thus the choice of K_V and K_P still needs to be reasonable, but importantly, will not affect the loop stability.

6.2.3 Determination of the Optimum Filter Prototype

In Section 6.2.1 the DPLL parameter values were determined using filter prototypes to place the loop poles. The prototype transfer function is given in equation (6.3), this is a generic solution where the constant values α, β, δ, ε and χ are selected for the particular class of prototype using Table 6.1 from earlier. In this section each filter prototype is used to determine the DPLL parameter values and the best performing prototype for the DPLL is found.

The DPLL system lock time t_{LCK} for each class of filter prototype is plotted using the piecewise linear stability methodology (using the piecewise linear methodology equation (5.41)). This is shown in Figure 6.3 below (blue line), the system steady state error is also included (red line).

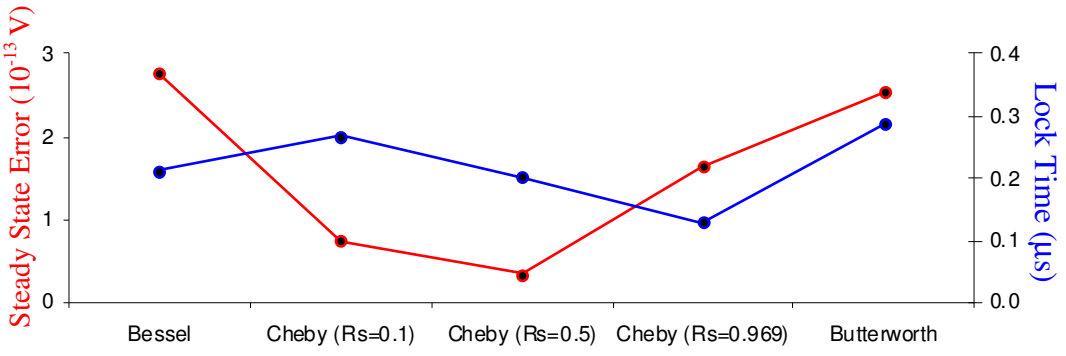

Figure 6.3 Lock Time and Steady State Error for Filter Prototypes

By inspection of the two plots in Figure 6.3 it can be seen that the Chebyshev filter returns the best results for both lock time and steady state error for all considered Chebyshev ripple parameters, R_s.

As an example, consider a DPLL system with gain parameters chosen as K_V = 62.8MHz/V and I_P = 10µA. The value of R_s allows the designer to trade-off between faster lock time and better steady state error. As R_s is increased ω_c approaches ω_R, this is illustrated in Figure 6.4 below.

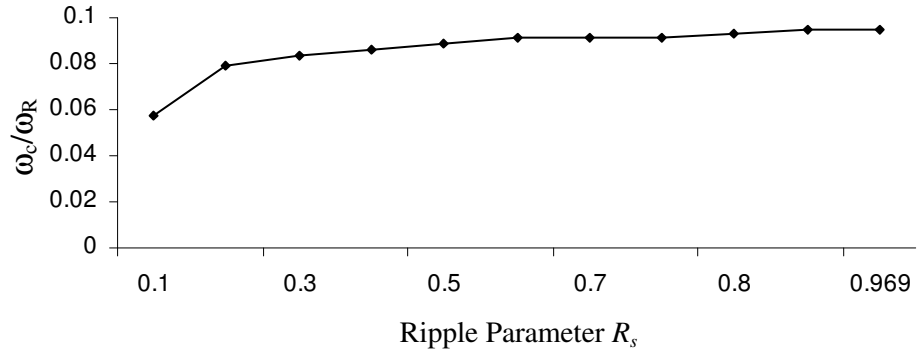

Figure 6.4 ω_C/ω_R for Range of R_s Parameter

The lock time and steady state error can be varied by optimally choosing R_s. Figure 6.5 again shows the lock time (blue line) and steady state error (red line) for a range

of R_s. The minimum lock time and steady state error is found to occur at a value of R_s equal to 0.707.

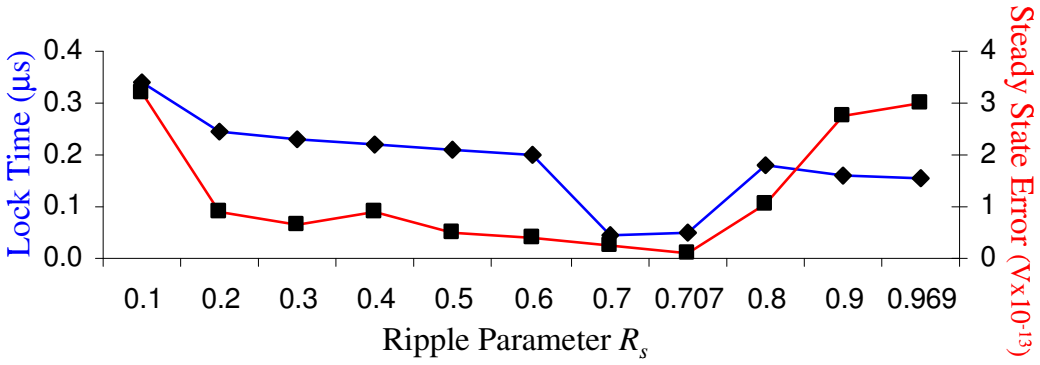

Figure 6.5 Lock Time (blue line, diamonds) and Steady State Error (red line, squares) for Chebyshev Filters

Thus by inspection of the results illustrated in Figure 6.5, and the results shown earlier in Figure 6.3, it can be concluded that the optimum filter prototype to determine the DPLL filter parameters is the Chebyshev type 1 class of prototype with a ripple parameter chosen to be 0.707. This prototype is applied exclusively from this point forward.

6.2.4 Placing the Fourth Order Pole

In Section 6.3.1 the two ratios M_1 and M_2 were introduced to help solve the fourth order prototype parameters. The parameters M_1 and M_2 both define the location of the fourth order filter pole P_4 in Figure 6.6.

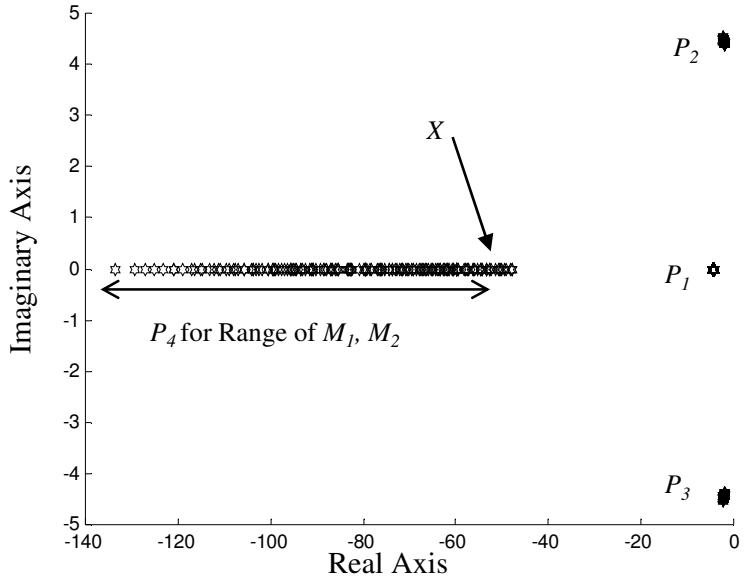

Figure 6.6 Pole Locations for Range of M_1 and M_2

It is desirable that all the loop poles are located in the optimum location as defined by the filter prototype poles; this was the purpose of equating both transfer functions earlier. However this is not feasible for the passive transconductance class of filter that is employed in the DPLL. This is because there is an inherent requirement for at least one real pole. The fourth order filter prototype has two complex conjugate pairs of poles and no real pole; this cannot be the case with the transconductance filter. The poles however can be placed as close to the ideal location by optimally choosing the parameter values M_1 and M_2.

The root locus plot of Figure 6.6 also shows that increasing M_1 and M_2 causes pole P_4 to move and P_1, P_2 and P_3 to remain relatively constant. Similarly Figure 6.7 shows that varying M_1 and M_2 affects the roll-off of the system magnitude. Ideally we require a value of M_1 and M_2 that will produce the sharpest roll-off, and therefore best filter cut-off characteristics. This occurs at the inflection point of P_4 where it is closest to the imaginary axis.

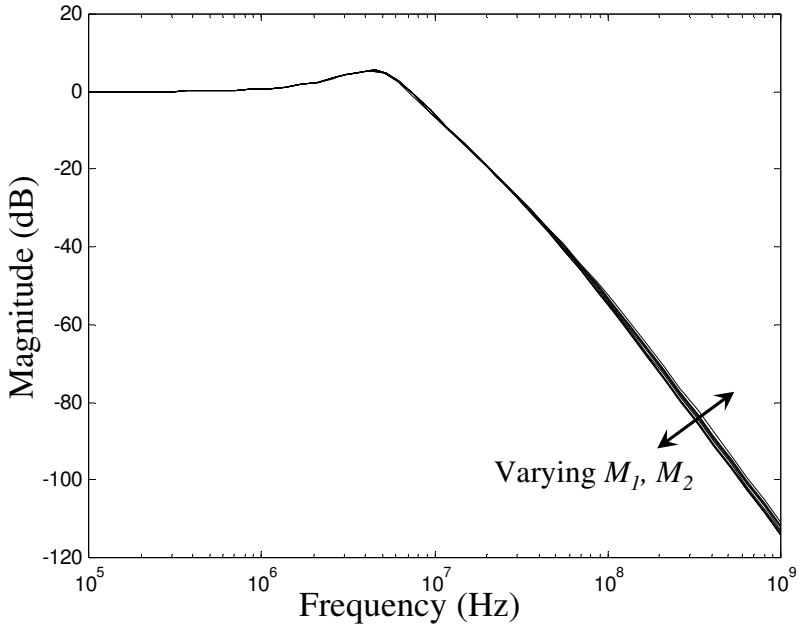

Figure 6.7 Bode Magnitude Plot for Range of M_1 and M_2

The trajectory of P_4 as M_1 and M_2 change is irregular, it initially moves closer to P_1 for increasing M_1 and M_2. When P_4 reaches the inflection point X, in Figure 6.6, the pole turns and moves away from P_1. The optimum choice of M_1 and M_2 is the point where P_4 lies at X – the pole location closest to the imaginary axis. To determine this inflection point exactly it is necessary to determine the roots of the system transfer function denominator. The denominator is a quartic equation and is therefore not easily solved; however the inflection point can be estimated using the following optimisation technique. The quartic PLL transfer function denominator can be expressed in terms of the individual poles as in equation (6.26) below; i.e. it has four poles placed in the left half plane with locations defined by the parameters *a, b, c,* and *d*.

$$D(s) = (s-a)(s-b)(s-c)(s-d) \tag{6.26}$$

The fourth order pole P_4 reaches inflection point X when the parameters *a, b, c,* and *d* are at a minimum. Multiplying these poles together the denominator can be determined in polynomial form as:

$$D(s) = s^4 - (a+b+c+d)s^3 + (ab+ac+ad+bc+bd+cd)s^2 - (abc+abd+acd+bcd)s + (abcd) \tag{6.27}$$

If equation (6.27) is expressed as in (6.28) then the minimum values of *a, b, c,* and *d* occur when the parameters *A*, *B*, *C* and *D* are minimised. In other words the inflection point of *A*, *B*, *C* and *D* corresponds to the inflection point of pole P_4.

$$D(s) = s^4 + As^3 + Bs^2 + Cs + D \tag{6.28}$$

The values of *A, B, C,* and *D* can be determined in terms of M_1, M_2 and the filter prototype parameters, by equating (6.28) with the denominator of the PLL transfer function in equation (6.2) and substituting in the loop filter component values, as defined in equations (6.10 – 6.12), (6.17) and (6.19). Thus the transfer function denominator coefficients are given in Table 6.2 below.

A	$\frac{\varepsilon\omega_c(M_1+1+M_1M_2)}{\delta}\left[1+\frac{\beta\delta(M_1M_2+1)-\alpha\varepsilon M_1M_2}{\alpha\varepsilon(M_1^2M_2+1)-\beta\delta(M_1M_2+1)+\alpha\varepsilon M_1M_2}\right]$
B	$\frac{\varepsilon^2\omega_c^2 M_1M_2}{\delta^2}\left[\frac{\left(M_1(1+M_2)+M_2+2+\frac{1}{M_1}\right)}{\frac{\alpha\varepsilon(M_1^2M_2^2+M_2)}{\beta\delta(M_1M_2+1)-\alpha\varepsilon M_1M_2}-M_2}+1\right]$
C	$\frac{M_1+1+M_1M_2}{\frac{\alpha}{\varepsilon\omega_c^3}-\frac{\beta\delta(M_1M_2+1)-\alpha\varepsilon M_1M_2}{\varepsilon^2\omega_c^3(M_1^2M_2+1)}}$
D	$\frac{\varepsilon^2\omega_c^2(M_1+1+M_1M_2)}{\frac{\alpha\delta}{\omega_c^2}-\frac{\beta\delta^2(M_1M_2+1)-\alpha\delta\varepsilon M_1M_2}{\varepsilon\omega_c^2(M_1^2M_2+1)}}$

Table 6.2 PLL Transfer Function Denominator Polynomial Coefficients

Using these parameters it is possible to approximate the inflection point of P_4 and thus choose the optimum values for M_1 and M_2. The inflection point is the same for

each of these parameters (*A, B, C,* and *D*); thus any one of the above parameters may be used to determine *X*.

As an example, consider a 1 GHz system with charge pump and VCO gains of 10μA and 125 Mrad/s/V respectively. The loop filter parameters are chosen using the filter prototype equations given in (6.10 – 6.12), (6.17) and (6.19). The optimum choice of M_1 and M_2 can be determined by substituting all known values into the four parameters in Table 6.2 and plotting any one of these for a range of M_1 and M_2. The *A* parameter is used to plot Figure 6.8 below.

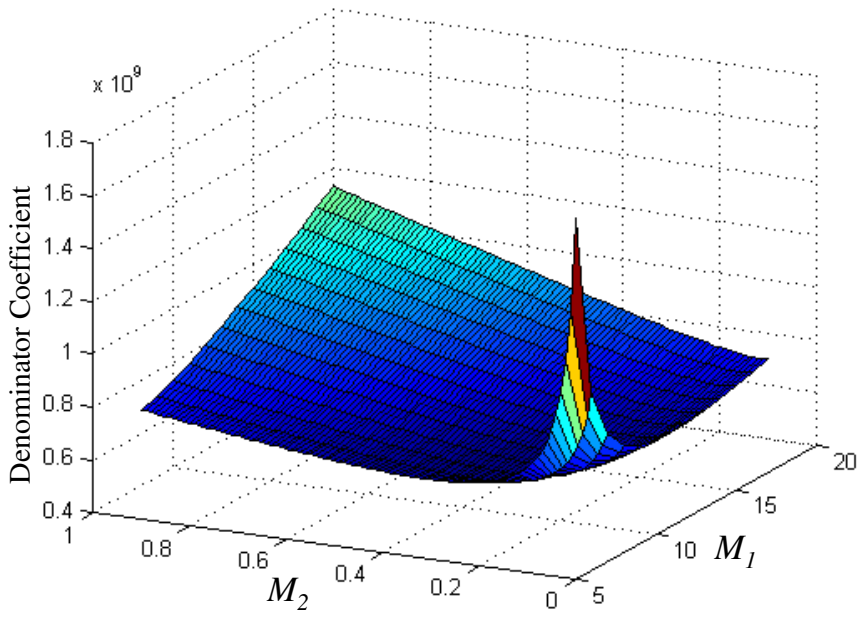

Figure 6.8 Determination of M_1 and M_2 Parameters for Optimum Pole Location. Plot of *A* Parameter.

The inflection point occurs at the minimum of *A* (or of any of the parameters in Table 6.2); resulting in the optimum values of M_1 and M_2. From the plot the optimum choice for M_1 and M_2 are found to be approximately 10 and 0.2 respectively. Finally, using the values of M_1 and M_2, loop filter parameter values can again be determined using (6.10 – 6.12), (6.17) and (6.19). The pole/zero plot and the frequency response are shown in Figure 6.9 and Figure 6.10.

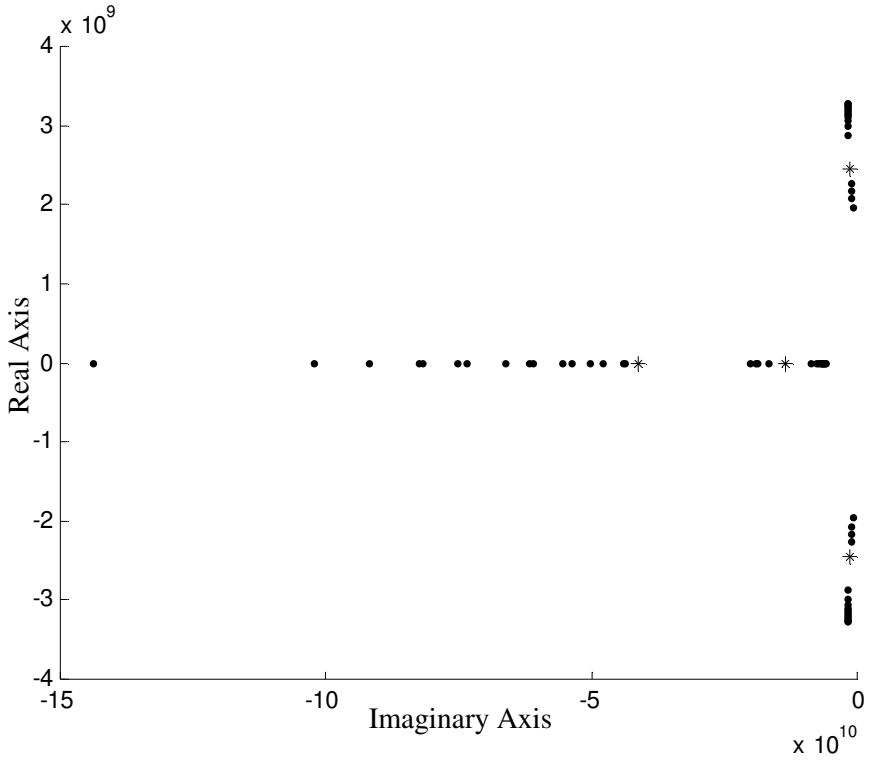

Figure 6.9 Pole/Zero Plot for a Range M_1 and M_2

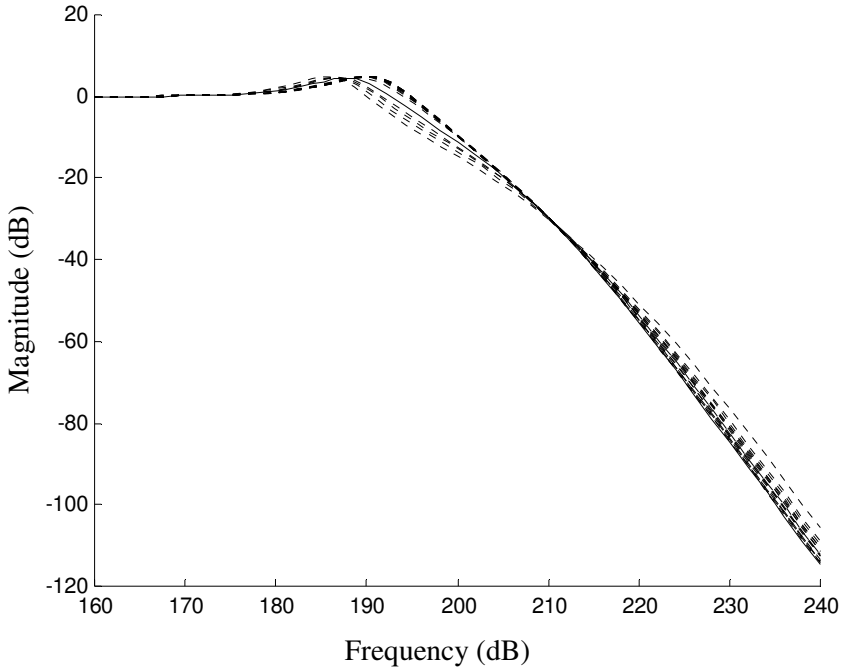

Figure 6.10 Frequency Response for a Range M_1 and M_2

In Figure 6.9 the 'stars' are the expected pole locations for the system with an M_1 equal to 10 and an M_2 equal to 0.2, as determined above. The 'dots' are the pole locations for systems with all other values of M_1 and M_2. As expected the selected system has the optimum fourth order pole location placed closest to the imaginary axis. This system is also expected to produce the best possible frequency cut-off characteristics; Figure 6.10 shows that this is the case. The frequency response for the designed system is the continuous line. The dashed lines are the responses for systems designed with a range of M_1 and M_2.

6.2.5 Determination of the Optimum Loop Bandwidth using the Piecewise Linear Stability Methodology

In this section, the piecewise linear stability methodology of the previous section is used to plot the system performance for a range of system bandwidths ω_c, where the bandwidth is chosen using the parameter values as defined by the filter prototype above. In other words by choosing ω_c, the DPLL parameters are calculated using equations (6.10 – 6.12), (6.17) and equation (6.19), then these parameter values are substituted into the piecewise linear equations as given in Appendix D. By iterating the piecewise linear stability equation the lock time of the DPLL can be determined for a range of system bandwidths ω_c as shown in Figure 6.11.

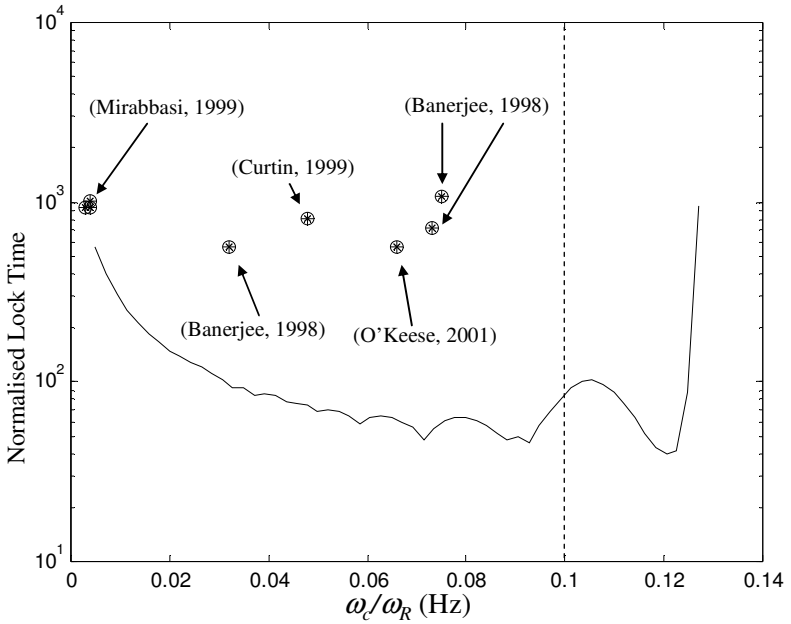

Figure 6.11 Plot of Traditional DPLL ω_C against Lock Time

In Figure 6.11 the system bandwidth is increasing from left to right on the plot. The continuous time approximation breakdown point (discussed earlier in Chapter 3) is to the right hand side of the plot. This is the point where ω_c is equal one tenth of ω_R and beyond this point the loop bandwidth becomes too small with the undesired affect of attenuating the information signal of the loop. From the earlier discussion on this topic, the suggestion that ω_c should be no greater than one tenth of ω_R, seems reasonable in light of the above observations.

The plot given in Figure 6.11 can be used to design the DPLL using the following steps:

1. Determine the DPLL system parameters using the prototype equations derived above ((6.10 – 6.12), (6.17) and equation (6.19)) for a range of ω_c.

2. The expected system lock time can then be plotted using the piecewise linear methodology of the previous section, as in Figure 6.11.

3. Finally, by choosing the optimum ω_c for the minimum lock time, using Figure 6.11, and choosing optimum M_1 and M_2 for the sharpest roll-off, a stable, and realizable fourth order DPLL can be designed.

Figure 6.11 also shows the normalised locations of traditionally designed systems using the design techniques of [13, 24-26, 43]. As can be seen, these methods provide slower lock time compared to the piecewise linear method for a similar choice of ω_c.

As a design example consider the design of 1 GHz fourth order DPLL system using the proposed methodology. This system has a feedback divide ratio of 1, and gains K_V = 377 MHz/V and I_P = 1mA. The optimum choice of M_1 and M_2 are found by plotting parameter A from Table 6.2 for a range of M_1 and M_2 and then choosing the values that produce the smallest output, see Figure 6.12.

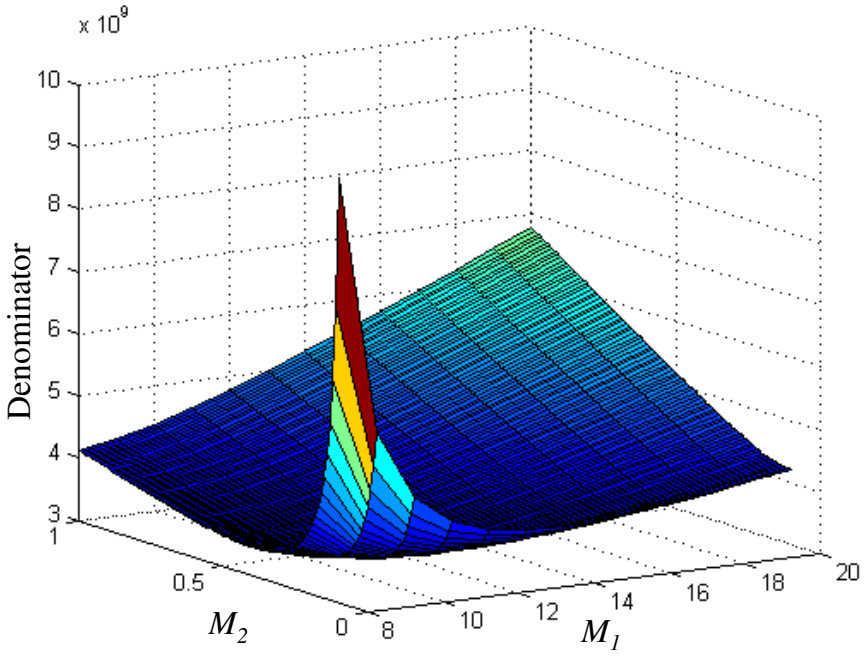

Figure 6.12 Plot of the Transfer Function Denominator Parameter for a Range of M_1 and M_2.

From the plot the minimum point (and therefore then inflection point of pole P_4) occurs at values of M_1 equal to 10 and M_2 equal to 0.2. Using these values of M_1 and M_2 and the piecewise linear methodology it is possible to plot the system lock time for a range of filter cut-off frequencies as in Figure 6.13.

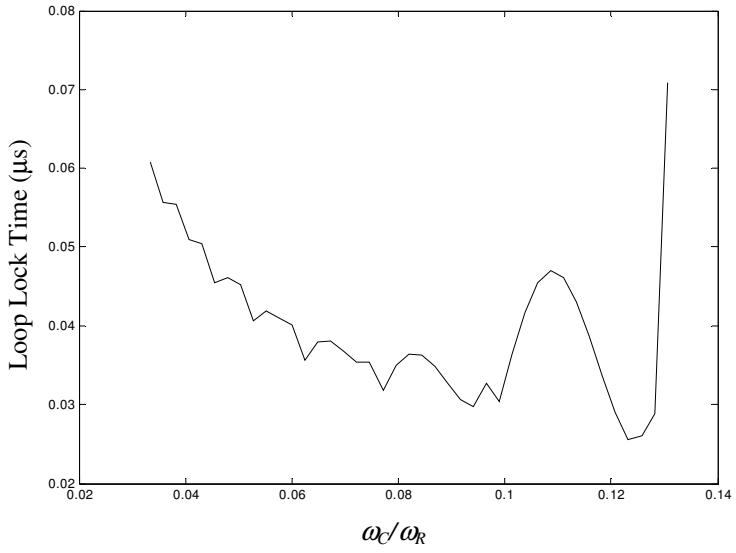

Figure 6.13 Plot of the the Piecewise Linear Model Lock Time for a Range of Filter Cut-off Frequencies

From Figure 6.13 the ω_C/ω_R is arbitrarily chosen to be 0.06. This results in the following set of loop filter parameter values: C_2 =2.4pF, C_3 = 0.24pF, C_4 = 40fF, R_2 = 2.9kΩ and R_4 = 14.6kΩ. Finally the circuit level simulated response of this system is plotted in Figure 6.14.

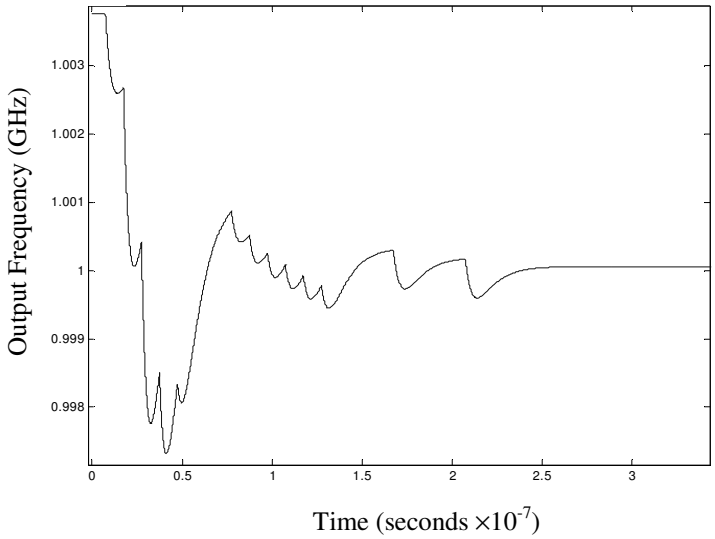

Figure 6.14 System Response of a 20MHz 4th Order DPLL

6.2.6 System Parameter Robustness

The filter prototype design technique outlined in the previous section determines realizable values for each of the DPLL loop filter components. However the necessary component values may need to be rounded or may suffer inaccuracies due to manufacturing. In this section the robustness of the system is considered as the component values are varied from their ideal value. To determine robustness, the filter component values are varied randomly by ±30% from the ideal values defined by equations (6.10 – 6.12), (6.17) and equation (6.19), and the outcome is considered. Consider a 1 GHz system, with such a large deviation (±30%.), the system poles are expected to move significantly, as shown in Figure 6.15 below.

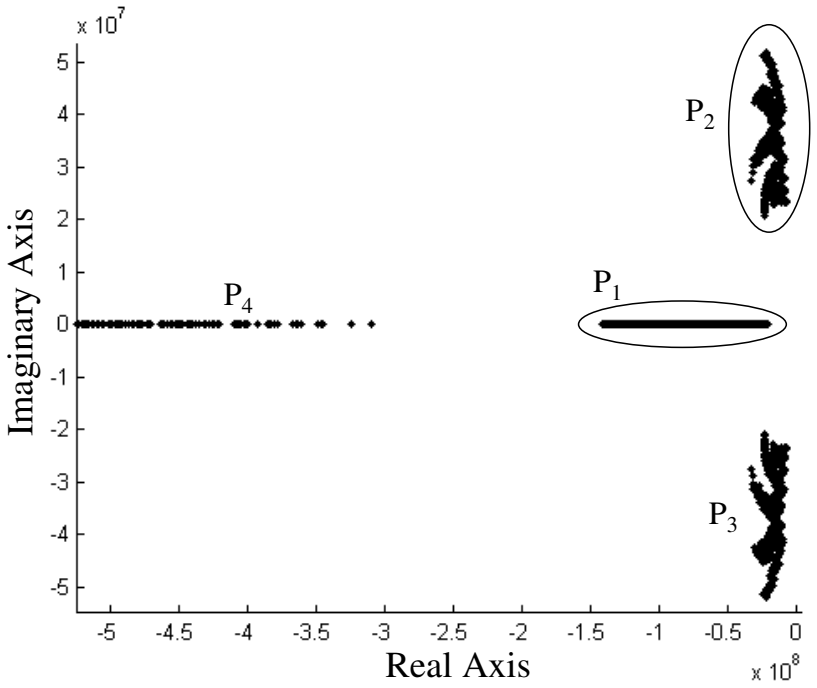

Figure 6.15 Pole Locations for Varying Component Offset.

There are three dominant poles affecting the system response, these are poles P_1 from Figure 6.15, and the two complex conjugate pair of poles P_2 and P_3. It is reasonable to conclude that any variation in the component values results in a corresponding movement of the system pole. The area of each ellipse in Figure 6.15 increase in size as the component percentage variations are increased. Stability, in the classical linear sense, means that all poles must lie to the left of the imaginary axis as illustrated in Figure 6.16 below. However, stability can be defined as low oscillatory response and rapid transient response, as described earlier at the beginning of Section 5.2.

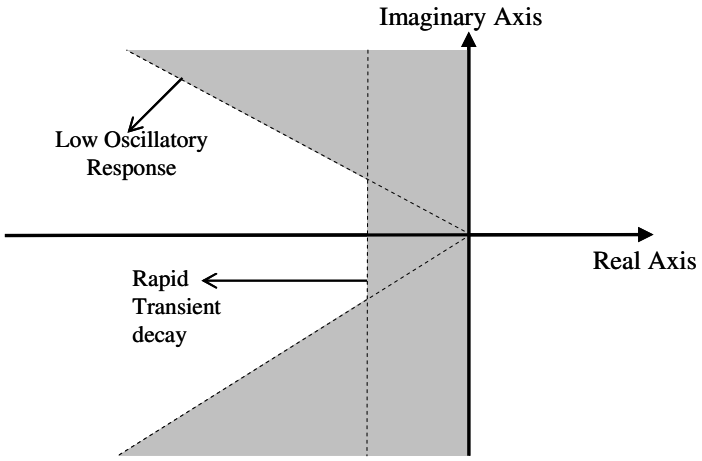

Figure 6.16 Combined Requirements for Continuous Systems

No single pole has a dominant effect on the system response; it is shared by the three poles, not including P_4. The system becomes less stable as the system poles approach the shaded area of Figure 6.16. The angle of the diagonal boundary in Figure 6.16 is typically chosen to be 45°, which gives a damping ratio of 0.707 [62, 63]. If we consider the system response in the frequency domain, Figure 6.17, rather than the pole movement, we should get a linear relationship between the cut off frequency offset from the ideal and the component variations.

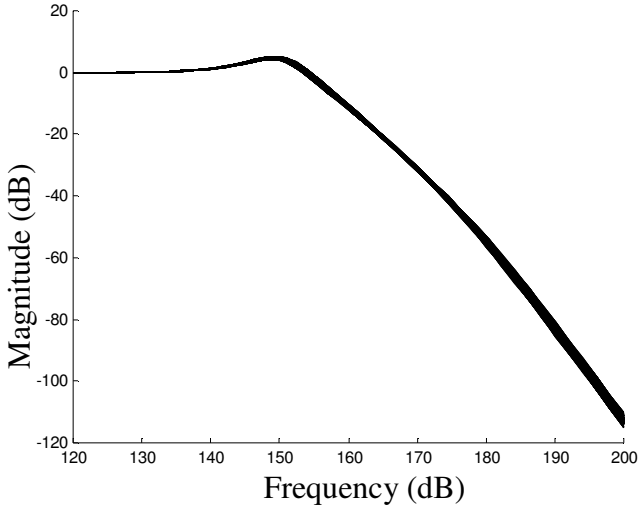

Figure 6.17 Plot of Frequency Response for a Range of Component Percentage Offsets

Figure 6.17 considers the –3db cut off frequency for positive and negative variations from the ideal, as the component values are varied. Using these deviations in the cut-

off frequency and plotting them against the loop filter parameter error, as in Figure 6.18, it can be seen that the system is stable to the left hand side of the dashed line and unstable otherwise.

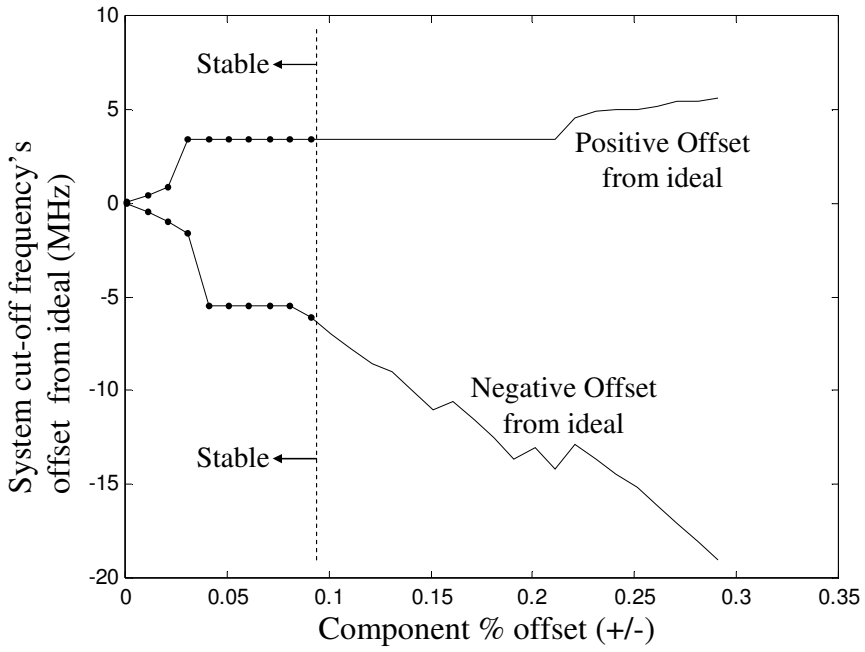

Figure 6.18 Plot of Cut-off Frequency Offset for a Range of Component Offsets.

From Figure 6.18 we can see that the system produces no unstable results for values of component offsets of less than 10%. In other words if all the components differ from the ideal by less than 10% the system will still be stable, but any further deviation may cause an unstable response. For a component variation of 9% the cut off frequency varies from –6.4 to +3.4 MHz. This is a cut-off frequency deviation range of 9.8MHz around the ideal frequency. Considering the reference signal operates at a frequency of 1 GHz this deviation is relatively small.

6.3 Design Methodology for the Fifth order DPLL

In this section, the same filter prototype procedure that was used in the previous section to determine optimum fourth order DPLL parameters is employed in order to determine the fifth order equivalent. As in the previous case, the filter prototype of

choice is the Chebyshev type 1 prototype (with a ripple parameter of 0.707), the fifth order Chebyshev transfer function is given in equation (6.29) below.

$$H_{\mathrm{Pr}ototype}(s) = \frac{\chi\omega_C^5}{s^5 + \alpha\omega_C s^4 + \beta\omega_C^2 s^3 + \delta\omega_C^3 s^2 + \varepsilon\omega_C^4 s + \chi\omega_C^5} \qquad (6.29)$$

where the parameter values are as given in Table 6.3 below.

Parameter	α	β	δ	ε	χ
Value	1.054	1.805	1.134	0.6594	0.1486

Table 6.3 Fifth Order Chebyshev Type 1 Parameter Values

The fifth order transfer function is of the form given in equation (6.1) with the relevant fourth order loop filter transfer function *F(s)* (equation (6.30)).

$$H_{5th}(s) = \frac{K_V K_P bs + K_V K_P a}{fs^5 + es^4 + ds^3 + cs^2 + \frac{K_V K_P b}{N} s + \frac{K_V K_P a}{N}} \qquad (6.30)$$

where the parameter values *a, b, c, d, e,* and *f* are given in Table 6.4 and $K_P = I_P/2\pi$.

Parameter	**Value**
a	1
b	R_2C_2
c	$C_2 + C_3 + C_4 + C_5$
d	$R_5C_2C_5 + R_4C_2C_4 + R_5C_4C_5 + R_2C_2C_4 + R_5C_3C_5 + +R_2C_2C_3 + R_4C_3C_4 + R_4C_2C_5 + R_4C_3C_5 + R_2C_2C_5$
e	$R_4R_5C_2C_4C_5 + R_2R_5C_2C_4C_5 + R_2R_4C_2C_3C_4 + +R_2R_5C_2C_3C_5 + R_4R_5C_3C_4C_5 + R_2R_4C_2C_3C_5$
f	$R_2R_4R_5C_2C_3C_4C_5$

Table 6.4 Fifth Order DPLL Transfer Function Parameters

The derivation of the fifth order DPLL prototype parameters is expansive and for that reason it is included in Appendix E. The derivation is carried out in the same manner as the fourth order DPLL in Section 6.2.1, except that in this case it is necessary to define four ratio parameters as given in equations (6.31 – 6.34).

$$M_1 = \frac{C_2}{C_3} \tag{6.31}$$

$$M_2 = \frac{C_4}{C_5} \tag{6.32}$$

$$N_1 = \frac{R_2}{R_4} \tag{6.33}$$

$$N_2 = \frac{R_2}{R_5} \tag{6.34}$$

Using these parameters and equating the fifth order prototype function of equation (6.29) with the fifth order PLL transfer function of equation (6.30), the DPLL parameters can be determined using the follow set of equations:

$$R_2 = \frac{\varepsilon}{\chi \omega_C C_2} \tag{6.35}$$

$$R_4 = \frac{\varepsilon}{N_1 \chi \omega_C C_2} \tag{6.36}$$

$$R_5 = \frac{\varepsilon}{N_2 \chi \omega_C C_2} \tag{6.37}$$

$$C_3 = \frac{C_2}{M_1} \tag{6.38}$$

$$C_5 = \frac{C_4}{M_2} \tag{6.39}$$

$$C_4 = \sqrt{C_2 \mu_1} \tag{6.40}$$

$$C_2 = \frac{\mu_1 \left(\sqrt{\left(1+\frac{1}{M_2}\right)^2 + 4\left(1+\frac{1}{M_1}\right)\mu_2} - \left(1+\frac{1}{M_2}\right) \right)^2}{4\left(1+\frac{1}{M_1}\right)^2} \tag{6.41}$$

where $\mu_1 = \dfrac{K_V K_P M_1 N_1 M_2 N_2 \chi^2}{\omega_C^2 \varepsilon^3}$ and $\mu_2 = \dfrac{\delta \varepsilon^3}{M_1 M_2 N_1 N_2 \chi^3}$

Using the above set of DPLL system parameters, it is possible to place the loop poles in the optimum location as defined by the filter prototype. In the case of the fifth order it is possible to have two complex conjugate pairs of poles and one real pole, thus placing all the system poles in the desired filter prototype arc.

Finally, by calculating the fifth order system parameters with the above equations for a range of loop bandwidths and then determining the system lock time using the piecewise linear stability methodology, it is possible to design stable fifth order DPLL systems with the desired lock time and an optimum loop bandwidth.

For example consider a fifth order 1GHz DPLL system, with a charge pump current gain I_P = 1mA, and a feedback divide ratio N = 4. Using the set of equations (6.35 – 6.41) and the piecewise linear stability methodology the expected system lock time for a range system bandwidths can be determine (Figure 6.19).

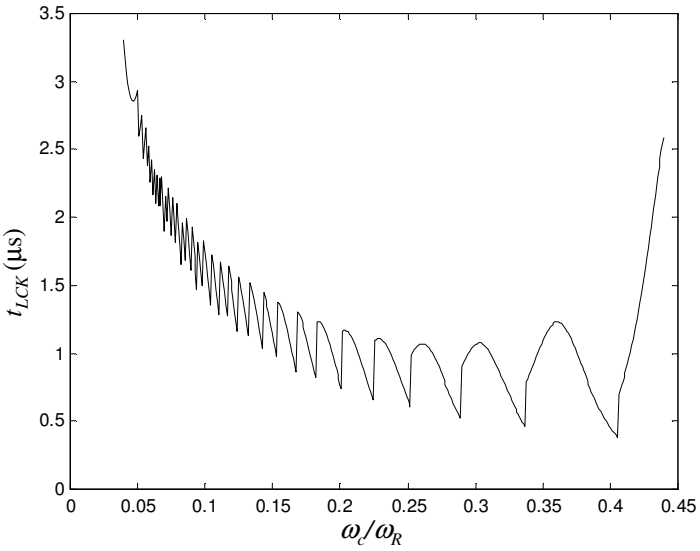

Figure 6.19 Plot of Expected System Lock Time for a Range of Bandwidths

From Figure 6.19 the loop filter bandwidth is chosen to be 200 MHz (ω_c/ω_R=0.2). Thus the system parameter values are K_V = 314 MHz/V, R_2 = 6.5kΩ, R_4 = 1kΩ, R_5 = 10kΩ, C_2 = 539fF, C_3 = 135fF, C_4 = 234fF, and C_5 = 58fF. A plot of the state space

response of this system using the piecewise linear methodology is as given in Figure 6.20. The expected pole locations for this system are given in Figure 6.21.

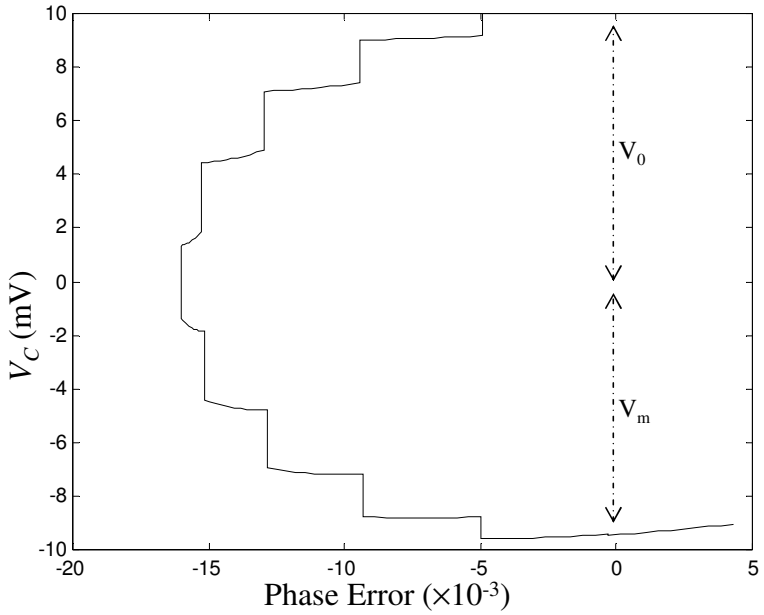

Figure 6.20 Plot of One Segment of the Fifth Order DPLL State Space Trajectory.

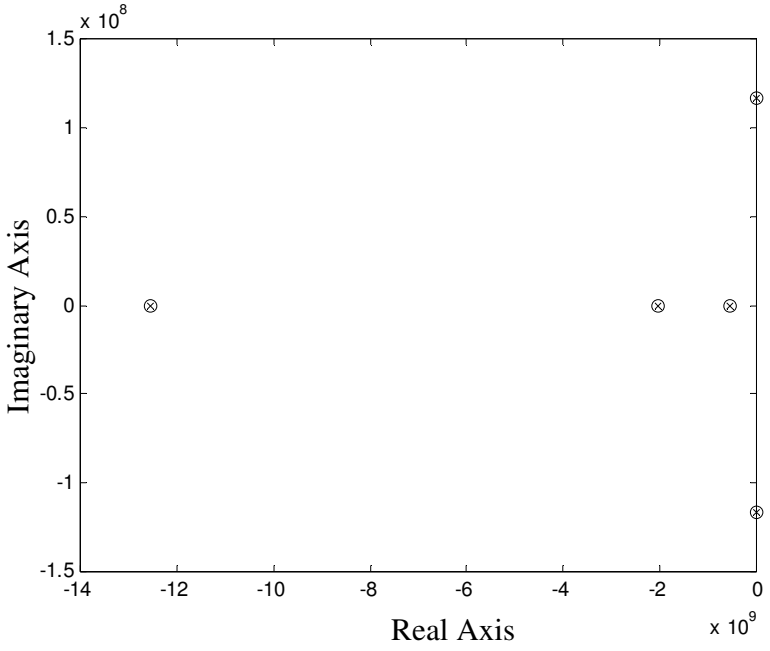

Figure 6.21 Plot of the Fifth Order System Pole Locations.

From the results of the piecewise linear state space plot this system is expected to be stable. This can be confirmed by simulating the above DPLL loop with the determined set of component values using a circuit level simulation. As expected it

is found that the system response is stable, see the transient response plot in Figure 6.22. The DPLL output frequency is 4GHZ, this is expected because the reference frequency is 1GHz and the feedback divide ratio is 4.

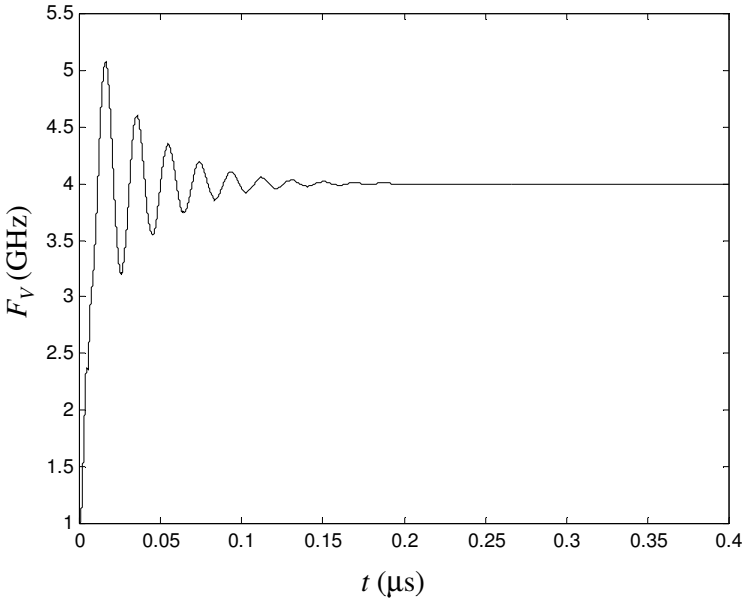

Figure 6.22 Circuit Level Simulation of the System Transient Response

6.4 Conclusion

In this chapter a high order DPLL design methodology was presented. This methodology combines filter prototype design methodologies and the piecewise linear methodology of Chapter 5 into one design technique. The former technique determines the loop filter parameter values by equating the denominator of the DPLL transfer function with that of the filter prototype; and the latter determines the stability boundary for this DPLL system for a predefined reference frequency and lock time. This design methodology is beneficial and desirable for the following reasons:

1. It reduces the number of system design parameters that need to be considered.

2. The result is a design methodology for high order DPLL systems where all the system poles are optimally placed, something that is not currently possible with existing methods.

3. The system transfer function was found to be independent of the VCO and CP gain parameters. The loop gain is set solely by the filter prototype and the choice of filter cut-off frequency.

4. The presented design methodology is not restricted to low orders, as is the case with existing methods.

It was found that although there are a number of different classes of filter prototypes, the best prototype for DPLL design is the Chebyshev (type 1) filter prototype with a choice of ripple parameter equal to 0.707.

Using the proposed methodology it is possible to plot the system lock time for a range of loop filter cut-off frequencies (loop bandwidth). This introduced a novel means of viewing the system stability. From these results it was shown that the continuous time approximation restriction (i.e. the reference frequency should be no greater than one tenth of the loop bandwidth) is a reasonable constraint. If this constraint is not adhered to the system would be highly unstable. In most cases in the literature this design constraint is taken literally, resulting in an excessively narrow loop bandwidth being placed excessively far away from the continuous time

approximation boundary. However by utilising the proposed design methodology this continuous time approximation boundary can be determined, thus enabling the designer to better choose the loop bandwidth, as close to the boundary as desired.

In traditional DPLL design methods the system poles are generally placed well away from the imaginary axis, this is more by accident (due to the rule-of-thumb process) than design. However the design method proposed in this chapter places the two most dominant poles quite close to the imaginary axis. This makes it necessary to test the design robustness to small variations in loop parameter values. Any parameter variations may cause the pole to move in to the right half plane, causing instability. The final system parameter values will not exactly match the desired values that were initially selected by the designer; this is due to slight inaccuracies in component values. However it was found in this section that for large deviations of parameter values the dominant poles did not move significantly and the stability of the DPLL system was maintained.

Finally, as an example a high order (fifth order) DPLL system was designed using the presented methodology. The lock time for this system was determined for a range of system bandwidths using the piecewise linear methodology. Choosing the optimum bandwidth the system parameters was then selected by the filter prototype methodology. Using traditional design methods stable high order DPLL design is a laborious task; in contrast the presented methodology designs stable systems using design methodologies that require only seconds to design and model.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

The DPLL system is traditionally analysed and designed using classical linear theory. The advantages of the linear approach are the modelling speed and availability of powerful analytical tools. Linear methods are found to give a good approximation of low order DPLL systems, however it is well documented [3, 14, 51], and has been illustrated in this thesis, that the linear approximation can not guarantee a globally stable prediction. This is due to the differences between the DPLL and the linear approximation, as the order is increased this inaccuracy becomes more substantial. For this reason, and the fact that the system analysis becomes more complex, high order DPLL systems are rarely used in practice. However the benefit of high order loops are improved out-of-band noise performance and therefore less frequency jitter on the DPLL output signal. These advantages cannot be ignored.

This thesis considers the issue of such high order DPLL systems and in particular their loop stability. The aim of this thesis is to enable high order system design. To achieve this, a number of issues needed to be addressed: the nonlinear nature of the DPLL loop; the complexity of the high order analysis; and the lack of a viable high order DPLL analysis tool. Each of these issues has been addressed in this thesis as follows:

1. The inherently nonlinear nature of the DPLL loop can be modelled using a piecewise linear approach to model the DPLL loop state transitions. This is achieved using behavioural modelling methods.

2. The complexity of the DPLL analysis is reduced by implementing a novel charge approximation of the loop filter equations. This has the beneficial effect of removing the high order system equation differential terms, thus a closed form model of the loop can be determined. This approach eases the high order loop restriction that would otherwise exist.

3. A novel piecewise linear stability methodology was presented; this simplified the DPLL analysis by considering the loops transient behaviour in state space. An early prediction of the system stability can be made by looking at the first few state space samples, this is achieved by determining whether the system state variables are converging to or diverging from the equilibrium that is phase lock.

4. For high order DPLL systems there are a large number of design parameters that need to be considered (seven in the case of the fourth order DPLL loop) – this is cumbersome. This can be improved by using filter prototypes and the new piecewise design methodology to select the DPLL parameter values, thus reducing the number of design parameters (for example three parameters are required to define the fourth order DPLL performance).

The resulting analysis and design methodology presented in this thesis is not mathematically restricted and can be implemented to arbitrarily high orders. An example of a fifth order design is given, this illustrates that the piecewise linear methodology can be used to accurately determine the response of the fifth order DPLL, something that is not feasible with existing methods.

The high order DPLL design and modelling methodology proposed in this thesis could be applied to other mixed signal systems such as the continuous time sigma-delta modulator (SDM). Both the DPLL and SDM are feedback loops containing both analogue and digital signals. This, however, has not been investigated in this thesis and has been left as future work.

There is currently a significant amount of research being carried out into the design of advanced architectures for DPLLs; some of these have already been discussed. One of these is a technique called ‘gear-shifting’; this is essentially the reducing of the DPLL loop bandwidth (in steps) as it approaches lock and increasing it if lock is lost. This results in a dynamic DPLL loop that is both fast locking and low noise. The bandwidth can be varied in a number of ways:

1. By varying the system gain; varying the loop filter bandwidth.

2. By changing the loop filter order.

In the case of 'gear-shifting' through loop orders, it is necessary to be able to determine stable high order system parameters; this is not currently achievable with existing design methods. However by utilising the design methodology proposed in this thesis, stable high order DPLL systems can be determined. When unlocked the loop filter will be of low order, once locked the loop will be 'gear-shifted' up to high order. In the case of the high order loop it will only be active when the loop is locked (phase error is small). The high order loop will only operate with a small phase error. For this reason any analysis of the system that is made using the proposed design methodology is expected to be accurate (as the model error is proportional to the phase error). While this presents another good application for the piecewise linear stability methodology, it has not been investigated as part of this thesis and has been left as future work.

In conclusion this thesis set out to accurately and efficiently model and analyse high order DPLL systems. This is a laborious and sometimes impossible task to achieve using traditional linear and nonlinear methods. It was found that high order DPLL modelling and analysis could be achieved by utilising three novel approaches: first by using a loop filter charge approximation to simplify the system equations; second, by taking a piecewise linear approach to the system model – accurately determining the system response; and finally by using filter prototypes to place the system poles – further simplifying the high order design procedure.

APPENDIX A

DESIGN OF HIGH ORDER DPLL FILTERS USING TWO-PORT METHODOLOGY

In this appendix arbitrary order DPLL filters are designed using two different low order Filter architectures. These are cascaded together to produce a single filter of high order. Using two ports circuit design theory, the DPLL system transfer function can be derived by multiplying the two filter transfer functions together. By cascading additional filters, and multiplying additional two port equations to the original, the filter equations can be efficiently increased to arbitrary order.

DPLL Filter Structure

A third order DPLL filter architecture is generally of the format shown in Figure A.1 below. This is a transconductance passive low pass filter that converts the incoming current information to a voltage output for operation by the VCO.

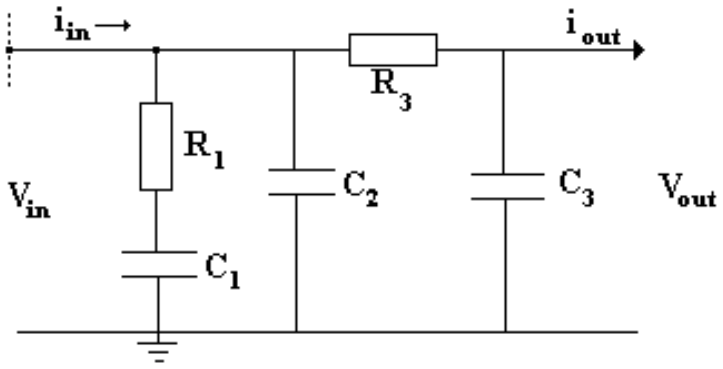

Figure A.1 Third Order Filter Structure

This can be split into two different filters, one of second order, Figure A.2 and one of first order, Figure A.3. By cascading additional filters similar in architecture to Figure A.3, filters of any order can be designed.

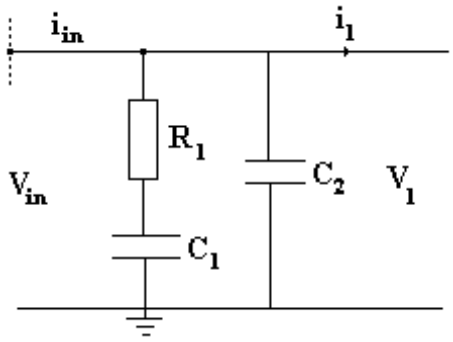

Figure A.2 Second Order Filter Structure

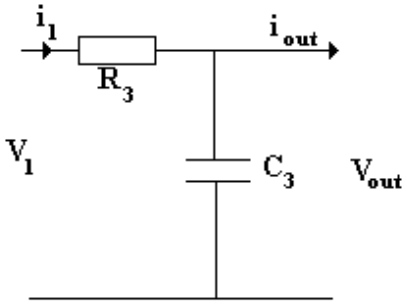

Figure A.3 High Order Filter Cascade

For this to work we need a two port mathematical model to represent each of these models. This enables the multiplication of each filters two port transfer function to give an overall transfer function for both filters.

- System 1

  The system in Figure A.2 has the following two port equation:

$$\begin{bmatrix} V_{in} \\ I_{in} \end{bmatrix} = \begin{bmatrix} \dfrac{C_1 C_2 R_1 s^2 + (C_1 + C_2)}{C_1 R_1 s + 1} & 0 \\ 1 & \end{bmatrix} \begin{bmatrix} V_1 \\ -I_1 \end{bmatrix} \qquad \text{(A.1)}$$

- System 2

  Similarly for the system of Figure A.3 has the two port equation:

$$\begin{bmatrix} V_1 \\ I_1 \end{bmatrix} = \begin{bmatrix} 1+C_3R_3s & R_3 \\ C_3s & 1 \end{bmatrix} \begin{bmatrix} V_{out} \\ -I_{out} \end{bmatrix} \tag{A.2}$$

To calculate the transfer function of the overall system, as shown in Figure A.1, the two port matrices in equation (A.1) and (A.2) are multiplied together.

To create a higher order DPLL additional filters are cascaded, so the transfer function can be calculated by multiplying equation (A.1) by the required amount of equation (A.2).

Example

The transfer function of Figure A.1 is calculated using equations (A.1) and (A.2). This is equal to:

$$\begin{bmatrix} V_{in} \\ I_{in} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \dfrac{C_1C_2R_1s^2+(C_1+C_2)}{C_1R_1s+1} & 1 \end{bmatrix} \begin{bmatrix} 1+C_3R_3s & R_3 \\ C_3s & 1 \end{bmatrix} \begin{bmatrix} V_{out} \\ -I_{out} \end{bmatrix} \tag{A.3}$$

When this matrix equation is solved we find that:

$$\frac{V_{out}}{I_{in}} = \frac{1+C_1R_1s}{\left(\begin{matrix} C_1C_2C_3R_1R_3s^3+(C_1C_3R_3+C_1C_2R_1+C_2C_3R_3+C_1C_3R_1)+ \\ +(C_1+C_2+C_3) \end{matrix}\right)} \tag{A.4}$$

This is the transfer function of the filter structure of Figure A.1.

Conclusion

Filters can be cascaded to create high order filters, the transfer function of this system is found by multiplying the relevant two port equations together. Finally the PLL linear transfer function is found by multiplying the filter transfer function *F(s)* into:

$$H(s) = \frac{K_V I_p F(s)}{2\pi s + \frac{K_V I_p F(s)}{N}} \tag{A.5}$$

where K_V is the VCO gain, I_P is the PFD charge pump gain, and N is the DPLL loop feedback divide ratio.

APPENDIX B

DERIVATION OF THE PHASE ERROR ZERO CROSSING CONTROL VOLTAGE

In this appendix the derivation of the second order DPLL control voltage in closed from is derived from equation (B.1).

$$V_m = V_0 + A\sum_{i=0}^{m-1}\sum_{k=0}^{i-1} B^{i-j-1}V_C(k) \qquad \text{(B.1)}$$

where $A = -K_V I_P T^2/C_2$ and $B = 1-(K_V R_2 I_P T))$. First consider the first few iterations of this equation as given in Table (B.1) below.

Iteration	Control Voltage
0	$V_0 = V_0$
1	$V_1 = V_0$
2	$V_2 = V_0 + AV_0$
3	$V_3 = V_0 + A\left(V_0\left(B+2\right)\right)$
4	$V_4 = V_0 + AV_0\left(B^2+2B+3\right) + A^2V_0$
5	$V_5 = V_0 + AV_0\left(B^3+2B^2+3B+4\right) + A^2V_0\left(2B+3\right)$
6	$V_6 = V_0 + AV_0\left(B^4+2B^3+3B^2+4B+5\right) + A^2V_0\left(3B^2+6B+6\right) + A^3V_0$
7	$V_7 = V_0 + AV_0\left(B^5+2B^4+3B^3+4B^2+5B+6\right) + A^2V_0\left(4B^3+9B^2+12B+10\right) + A^3V_0\left(3B+4\right)$
8	$V_8 = V_0 + AV_0\left(B^6+2B^5+3B^4+4B^3+5B^2+6B+7\right) + A^2V_0\left(5B^4+12B^3+18B^2+20B+15\right) + A^3V_0\left(6B^2+12B+10\right) + A^4V_0$

Table B.1 First Eight Iterations of the Control Voltage.

By looking at these first few iterations, it can be deduced that the double summation equation (B.1) can be replaced with equation (B.2).

$$V_m = V_0 \begin{pmatrix} 1 + A(\Lambda_1) \\ +A^2(\Lambda_2) \\ +A^3(\Lambda_3) \\ \vdots \\ +A^{\left\lceil \frac{m}{2} \right\rceil}\left(\Lambda_{\left\lceil \frac{m}{2} \right\rceil}\right) \end{pmatrix} \tag{B.2}$$

where Λ is a function of B and is defined as in Appendix C. For example to determine the 8th period of the control voltage V_8 it is necessary to determine all Λ parameters up to $\left\lceil \frac{8}{2} \right\rceil$ (i.e. Λ_1, Λ_2, Λ_3, and Λ_4). These are determined using the definitions of Λ given in Appendix C as follows:

1. The parameter Λ_1 can be calculated using equation (C.1), for a value of $n = 8$.

$$\Lambda_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}^T \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & B & 0 & 0 & 0 & 0 & 0 \\ 1 & B & B^2 & 0 & 0 & 0 & 0 \\ 1 & B & B^2 & B^3 & 0 & 0 & 0 \\ 1 & B & B^2 & B^3 & B^4 & 0 & 0 \\ 1 & B & B^2 & B^3 & B^4 & B^5 & 0 \\ 1 & B & B^2 & B^3 & B^4 & B^5 & B^6 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \tag{B.3}$$

$$\Rightarrow \Lambda_1 = B^6 + 2B^5 + 3B^4 + 4B^3 + 5B^2 + 6B + 7 \tag{B.4}$$

2. Similarly using equation (C.2) Λ_2 can be determined:

$$\Lambda_2 = \begin{bmatrix} B^4+2B^3+3B^2+4B+5 \\ B^3+2B^2+3B+4 \\ B^2+2B+3 \\ B+2 \\ 1 \end{bmatrix}^T \times \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & B & 0 & 0 & 0 \\ 1 & B & B^2 & 0 & 0 \\ 1 & B & B^2 & B^3 & 0 \\ 1 & B & B^2 & B^3 & B^4 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \tag{B.5}$$

$$\Rightarrow \Lambda_2 = 5B^4 + 12B^3 + 18B^2 + 20B + 15 \tag{B.6}$$

3. Using (C.3) and noting that $\Lambda_3 = \Gamma_6$, then Λ_3 can be calculated as:

$$\Lambda_3 = \Gamma_6 = \begin{bmatrix} B^2+2B+3 \\ B+2 \\ 1 \end{bmatrix}^T \times \begin{bmatrix} 1 & 0 & 0 \\ 1 & B & 0 \\ 1 & B & B^2 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 1 & \frac{B+1}{B} & 0 \\ 1 & \frac{B+1}{B} & \frac{B^2+B+1}{B^2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{B.7}$$

$$\Rightarrow \Lambda_3 = 6B^2 + 12B + 10 \tag{B.8}$$

4. Λ_3 is calculated using (C.4):

$$\Lambda_4 = [1]^T [1][1] = 1 \tag{B.9}$$

5. Finally substituting all these Λ values back into to (B.3) the control voltage at the 8th period of the reference signal is estimated as:

$$\begin{aligned} V_8 = {} & V_0 + AV_0\left(B^6+2B^5+3B^4+4B^3+5B^2+6B+7\right) \\ & + A^2V_0\left(5B^4+12B^3+18B^2+20B+15\right) \\ & + A^3V_0\left(6B^2+12B+10\right) + A^4V_0 \end{aligned} \tag{B.10}$$

Note that equation (B.10) is the same results as determined in table B.1.

APPENDIX C

CLOSED FORM SOLUTION OF THE CONTROL VOLTAGE

This Section defines the parameters of Λ as used earlier in equations (5.31), (5.32) and (5.34), and in Appendix B.

$$\Lambda_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}^T \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & B & 0 & 0 & 0 \\ 1 & B & B^2 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 1 & B & B^2 & \cdots & B^{n-2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \tag{C.1}$$

$$\Lambda_2 = \begin{bmatrix} B^{n-4} + 2B^{n-5} + \cdots + (n-4)B + (n-3) \\ B^{n-5} + 2B^{n-6} + \cdots + (n-5)B + (n-4) \\ \vdots \\ B+2 \\ 1 \end{bmatrix}^T \times \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & B & 0 & 0 & 0 \\ 1 & B & B^2 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 1 & B & B^2 & \cdots & B^{n-4} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \tag{C.2}$$

If we consider Λ_3 to be equal to the function Γ_k given in equation (C.3), where $k = 6$, then Λ_4 can be calculated using equation (C.4).

$$\Gamma_k = \begin{bmatrix} B^{n-k} + 2B^{n-k-1} + \cdots + (n-k)B + (n-k+1) \\ B^{n-k-1} + 2B^{n-k-2} + \cdots + (n-k-1)B + (n-k) \\ \vdots \\ B+2 \\ 1 \end{bmatrix}^T$$

$$\times \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & B & 0 & 0 & 0 \\ 1 & B & B^2 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 1 & B & B^2 & \cdots & B^{n-k} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & \frac{B+1}{B} & 0 & 0 & 0 \\ 1 & \frac{B+1}{B} & \frac{B^2+B+1}{B^2} & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 1 & \frac{B+1}{B} & \frac{B^2+B+1}{B^2} & \cdots & \frac{B^{n-k}+\cdots+B+1}{B^{n-k}} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad \text{(C.3)}$$

$$\Lambda_4 = \begin{bmatrix} \Gamma_8 \\ \Gamma_9 \\ \Gamma_{10} \\ \vdots \\ \Gamma_n \end{bmatrix}^T \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & B & 0 & 0 & 0 \\ 1 & B & B_2 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & B & B_2 & \cdots & B_{n-8} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad \text{(C.4)}$$

It is possible to calculate all Λ up to $\lceil m/2 \rceil$ by using the same process between equations (C.2) and (C.4), i.e. define Γ'_k as in equation (C.5). Then Λ_5 can be calculated as in (C.6), and so on up to $\Lambda_{\lceil m/2 \rceil}$.

$$\Gamma'_k = \begin{bmatrix} \Gamma_k \\ \Gamma_{k+1} \\ \Gamma_{k+2} \\ \vdots \\ \Gamma_n \end{bmatrix}^T \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & B & 0 & 0 & 0 \\ 1 & B & B_2 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & B & B_2 & \cdots & B_{n-k} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad \text{(C.5)}$$

$$\Lambda_5 = \begin{bmatrix} \Gamma'_{10} \\ \Gamma'_{11} \\ \Gamma'_{12} \\ \vdots \\ \Gamma'_{n} \end{bmatrix}^T \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & B & 0 & 0 & 0 \\ 1 & B & B_2 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & B & B_2 & \cdots & B_{n-10} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \tag{C.6}$$

APPENDIX D

PIECEWISE LINEAR DPLL MODEL EQUATIONS

The third through fifth order piecewise linear model equations are outlined in the subsections below. In each of these cases, the control voltage V_C is determined from the charge on the filter capacitors, where W is the system order.

$$V_C(t_{k+1}) = \frac{Q_W(t_{k+1})}{C_W} \tag{D.1}$$

The integral of V_C, to replace the second order calculation in equation (5.3), is calculated from the knowledge of V_C as shown in equation (D.2).

$$\int V_C dt = T_P V_C(t_k) + \frac{I_W(t_{k+1}) T_P^2}{2C_Z} \tag{D.2}$$

where T_P is the length of time that the DPLL will remain in the present state (either T_B or T_C depending on whether the DPLL loop is in the Up or Null state respectively), and T_B is equal to $|\phi_e(k)/2\pi F_R|$ and $T_C = T\text{-}T_B$.

Third Order DPLL Behavioural Equations

Q_2 and Q_3, the charges on the loop filter capacitors, C_2 and C_3 respectively, are calculated as follows:

$$Q_2(t_{k+1}) = Q_2(t_k) - T_P(I_P - I_3(t_{k+1})) \tag{D.3}$$

$$Q_3(t_{k+1}) = Q_3(t_k) + T_P I_3(t_{k+1}) \tag{D.4}$$

where I_3 is calculated as:

$$I_3(t_{k+1}) = I_P - \frac{T_P^2 I_P C_2 + T_B C_2 C_3 (\frac{Q_2(t_k)}{C_2} - \frac{Q_3(t_k)}{C_3})}{T_B(C_2 + C_3) + R_2 C_2 C_3} \quad \text{(D.5)}$$

Fourth Order DPLL Behavioural Equations

For the fourth order system the charge on the loop filter capacitors are calculated as in equation (D.6 – D.8).

$$Q_2(t_{k+1}) = Q_2(t_k) + \frac{T_P^2 I_3 C_2 - T_P C_2 C_3 (\frac{Q_2(t_k)}{C_2} - \frac{Q_3(t_k)}{C_3})}{T_P C_3 + R_2 C_2 C_3} \quad \text{(D.6)}$$

$$Q_3(t_{k+1}) = Q_3(t_k) + T_P I_3 \quad \text{(D.7)}$$

$$Q_4(t_{k+1}) = Q_4(t_k) + \frac{T_P^2 I_3 C_2 - T_P C_2 C_3 (\frac{Q_2(t_k)}{C_2} - \frac{Q_3(t_k)}{C_3})}{T_P C_3 + R_2 C_2 C_3} + T_P I_3 - T_P I_P \quad \text{(D.8)}$$

where I_3 and I_4 are the current through C_3 and C_4 and are calculated as in equations (D.9) and (D.10).

$$I_3 = \frac{\dfrac{\left(\dfrac{Q_2(t_k)}{C_2} - \dfrac{Q_3(t_k)}{C_3}\right)}{R_2 + \dfrac{T_P}{C_2}} - \dfrac{\left(\dfrac{Q_3(t_k)}{C_3} - \dfrac{Q_4(t_k)}{C_4}\right)}{R_4 + \dfrac{T_P}{C_4}} + I_P}{\dfrac{C_2 T_P}{C_3 T_P + C_2 C_3 R_2} + \dfrac{C_4 T_P}{C_3 T_P + C_3 C_4 R_4} + 1} \tag{D.9}$$

$$I_4 = I_p - I_3 - \frac{\left(I_3 C_2 T_P\right) - \left(C_2 C_3 \left(\dfrac{Q_2(t_k)}{C_2} - \dfrac{Q_3(t_k)}{C_3}\right)\right)}{\left(C_3 T_P\right) + \left(C_2 C_3 R_2\right)} \tag{D.10}$$

Fifth Order DPLL Behavioural Equations

The fifth order loop filter capacitor charges are calculated as shown in equations (D.11 – D.14) given below.

$$Q_2(t_{k+1}) = Q_2(t_k) + I_2 T_P \tag{D.11}$$

$$Q_3(t_{k+1}) = Q_3(t_k) + \frac{I_2 C_3 (R_2 C_2 + T) + C_3 Q_2(t_k) - C_2 Q_3(t_k)}{C_2 T} \tag{D.12}$$

$$Q_4(t_{k+1}) = Q_4(t_k) - \frac{D_1}{D_2} + \frac{T(R_5 C_5 + T)\left(I_2 (R_2 C_2 + T) + C_2 T \left(\dfrac{Q_2(t_k)}{C_2} - \dfrac{Q_3(t_k)}{C_3}\right)\right)}{C_2 T D_2} \tag{D.13}$$

$$Q_5(t_{k+1}) = Q_5(t_k) + I_5 T_P \tag{D.14}$$

where I_2, I_5 and D_1 to D_4 are defined as in equations (D.15 – D.20).

$$I_2 = \frac{D_3 - D_4\left(\frac{Q_2(t_k)}{C_2} - \frac{Q_3(t_k)}{C_3}\right)}{1 + (R_2C_2 + T)\frac{D_4}{C_2}} \tag{D.15}$$

$$I_5 = \frac{\left[\begin{array}{c} C_5T^2(R_5C_5 + T)\left(I_2(R_2C_2 + T) + C_2T\left(\frac{Q_2(t_k)}{C_2} - \frac{Q_3(t_k)}{C_3}\right)\right) \\ -C_2C_5T^2D_1 + C_2C_5TD_2Q_4(t_k) - C_2C_4TD_2Q_5(t_k) \end{array}\right]}{C_2C_4TD_2(R_5C_5 + T)} \tag{D.16}$$

$$D_2 = \frac{(R_3C_3 + T)(R_4C_4 + T) + C_4R_3T}{C_3} \tag{D.17}$$

$$D_1 = \left(\frac{Q_3(t_k)}{C_3} - \frac{Q_2(t_k)}{C_2}\right)(R_4C_4 + T) + C_4R_3\left(\frac{Q_3(t_k)}{C_3} - \frac{Q_4(t_k)}{C_4}\right) \tag{D.18}$$

$$D_3 = \frac{D_1(C_3(R_4C_4 + T) + C_4T)}{D_2C_3(R_4C_4 + T)} - I_P - \frac{C_4\left(\frac{Q_3(t_k)}{C_3} - \frac{Q_2(t_k)}{C_2}\right)}{R_4C_4 + T} \tag{D.19}$$

$$D_4 = \frac{C_2C_3T + TC_3(R_4C_4 + T) + C_4T^2}{C_3TD_2} \tag{D.20}$$

APPENDIX E

DERIVATION OF FIFTH ORDER FILTER PROTOTYPE EQUATIONS

In this appendix the derivation of the fifth order DPLL design equations using filter prototypes is given. This derivation starts by equating the denominator of the fifth order DPLL transfer function given in equation (E.1), with the denominator of the fifth order normalised and generic prototype filter given in (E.2).

$$H_{5th}(s) = \frac{K_V I_P b s + K_V I_P a}{f s^5 + e s^4 + d s^3 + c s^2 + \frac{K_V I_P b}{N} s + \frac{K_V I_P a}{N}} \tag{E.1}$$

$$H_{\mathrm{Pr}ototype}(s) = \frac{\chi \omega_c^5}{s^5 + \alpha \omega_c s^4 + \beta \omega_c^2 s^3 + \delta \omega_c^3 s^2 + \varepsilon \omega_c^4 s + \chi \omega_c^5} \tag{E.2}$$

where the parameter values *a, b, c, d, e,* and *f* are given in Table E.1.

Parameter	Value
a	1
b	R_2C_2
c	$C_2 + C_3 + C_4 + C_5$
d	$R_5C_2C_5 + R_4C_2C_4 + R_5C_4C_5 + R_2C_2C_4 + R_5C_3C_5 +$ $+R_2C_2C_3 + R_4C_3C_4 + R_4C_2C_5 + R_4C_3C_5 + R_2C_2C_5$
e	$R_4R_5C_2C_4C_5 + R_2R_5C_2C_4C_5 + R_2R_4C_2C_3C_4 +$ $+R_2R_5C_2C_3C_5 + R_4R_5C_3C_4C_5 + R_2R_4C_2C_3C_5$
f	$R_2R_4R_5C_2C_3C_4C_5$

Table E.1 Fifth Order DPLL Transfer Function Parameters

By equating both parameters the following set of simultaneous equations (E.3 – E.7) are determined:

$$\chi\omega_c^5 = \frac{K_{sys}}{C_2C_3C_4C_5R_2R_4R_5} = \frac{K_{sys}}{f} \tag{E.3}$$

where K_{sys} is the system gain and is equal to K_VK_P.

$$\varepsilon\omega_c^4 = \frac{K_{sys}}{C_3C_4C_5R_4R_5} = \frac{bK_{sys}}{f} \tag{E.4}$$

$$\delta\omega_c^3 = \frac{C_2 + C_3 + C_4 + C_5}{C_2C_3C_4C_5R_2R_4R_5} = \frac{c}{f} \tag{E.5}$$

$$\beta\omega_c^2 = \frac{d}{f} \tag{E.6}$$

$$\alpha\omega_c = \frac{e}{f} \tag{E.7}$$

There are seven unknown parameters but only five simultaneous equations, thus it is necessary to define a number of additional ratio parameters as given in equations (6.30 – 6.33).

$$M_1 = \frac{C_2}{C_3} \tag{E.8}$$

$$M_2 = \frac{C_4}{C_5} \tag{E.9}$$

$$N_1 = \frac{R_2}{R_4} \tag{E.10}$$

$$N_2 = \frac{R_2}{R_5} \tag{E.11}$$

Using the nine simultaneous equations given above it is possible to determine the loop filter parameter values in terms of the filter prototype coefficients as follows:

1. Equation (E.4) can be rewritten as:

$$\frac{\varepsilon\omega_c^4}{a} = \frac{K_{sys}}{f} \tag{E.12}$$

Using (E.5) the left hand side of (E.12) is equal to:

$$\frac{\varepsilon\omega_c^4}{b} = \frac{K_{sys}}{f} = \chi\omega_c^5 \tag{E.13}$$

Since $b = R_2C_2$ this can be solved for R_2:

$$R_2 = \frac{\varepsilon}{\chi\omega_c C_2} \tag{E.14}$$

2. To calculate the parameter C_3 consider equation (E.3) above. Using the definitions given in equations (E.8 – E.11) this can be rewritten as:

$$\chi\omega_c^5 = \frac{K_{sys}}{C_2C_3C_4C_5R_2R_4R_5} = \frac{K_{sys}}{\frac{C_2^2C_4^2R_2^3}{M_1M_2N_1N_2}} \tag{E.15}$$

This can be reduced to:

$$\chi\omega_c^5 C_2^2 C_4^2 R_2^3 = K_{sys} M_1 M_2 N_1 N_2 \tag{E.16}$$

Substituting in the calculation of R_2 given in equation (E.14):

$$\frac{\chi\omega_c^5 C_2^2 C_4^2 \varepsilon^3}{\chi^3 \omega_c^3 C_2^3} = \frac{\omega_c^2 C_4^2 \varepsilon^3}{\chi^2 C_2} \triangleq K_{sys} M_1 M_2 N_1 N_2 \tag{E.17}$$

This can be solved for C_4 as follows:

$$C_4 = \sqrt{\frac{K_{sys}\chi^2 C_2 M_1 M_2 N_1 N_2}{\omega_c^2 \varepsilon^3}} \tag{E.18}$$

Letting $\mu_1 = \frac{KM_1 N_1 M_2 N_2 \chi^2}{\omega_C^2 \varepsilon^3}$ then (E.18) can be rewritten as:

$$C_4 = \sqrt{C_2 \mu_1} \tag{E.19}$$

3. All of the loop filter parameters have been determined in terms of the filter prototype coefficients and the filter capacitor C_2. The last step in this derivation is to determine C_2 in terms of the prototype coefficients but independent of the loop filter parameters. This can be achieved as follows.

Using equation (E.3) K_{sys} can be calculated as:

$$K_{sys} = \chi\omega_c^5 f \tag{E.20}$$

Substituting this in to (E.4) and solving for ω_c gives:

$$\omega_c = \frac{\varepsilon}{b\chi} \tag{E.21}$$

Substituting this into (E.5) gives:

$$\frac{\delta\varepsilon^3}{\chi^3}\frac{f}{b^3} = c \tag{E.22}$$

Substituting in the values of *b, c* and *f* from Table E.1 we get:

$$\frac{\delta\varepsilon^3}{\chi^3}\frac{C_2C_3C_4C_5R_2R_4R_5}{R_2^3C_2^3} = C_2 + C_3 + C_4 + C_5 \tag{E.23}$$

Again substituting in the known values for C_3, C_5, R_2, R_4 and R_5 using (E.8 – E.11) and (E.14):

$$\begin{aligned}\delta\varepsilon^3 C_4^2 = {} & C_2^2\chi^3 M_1M_2N_1N_2\left(1+\frac{1}{M_1}\right) \\ & + C_2C_4\chi^3 M_1M_2N_1N_2\left(1+\frac{1}{M_2}\right)\end{aligned} \tag{E.24}$$

Solving for C_2 gives the quadratic equation:

$$C_2^2\left(1+\frac{1}{M_1}\right) + C_2\left(C_4\left(1+\frac{1}{M_2}\right)\right) - \mu_2 C_4^2 = 0 \tag{E.25}$$

where $\mu_2 = \dfrac{\delta\varepsilon^3}{M_1M_2N_1N_2\chi^3}$.

Using the quadratic formula this can be solved by:

$$C_2 = \frac{-C_4\left(1+\frac{1}{M_2}\right) \pm \sqrt{C_3^2\left(1+\frac{1}{M_2}\right)^2 + 4\mu_2 C_4^2\left(1+\frac{1}{M_1}\right)}}{2\left(1+\frac{1}{M_1}\right)} \tag{E.26}$$

Substituting in C_4 from equation (E.19) and solving gives:

$$C_2 = \frac{\mu_1\left(\sqrt{\left(1+\frac{1}{M_2}\right)^2 + 4\left(1+\frac{1}{M_1}\right)\mu_2} - \left(1+\frac{1}{M_2}\right)\right)^2}{4\left(1+\frac{1}{M_1}\right)^2} \tag{E.27}$$

Thus the fifth order DPLL system loop filter parameters can be selected using the filter prototype coefficients and equations (E.8 – E.11), (E.14), (E.19) and (E.27).

REFERENCES

1. Best, R.E., *Phase Locked Loops*. 1984: McGraw-Hill.

2. Wang, Z., *An Analysis of Charge-Pump Phase-Locked Loops.* IEEE trans. on Circuits and Systems, 2005. **52**(10): p. 2128-2138.

3. Van-Paemel, M., *Analysis of a Charge-Pump PLL: A New Model.* IEEE trans. on Communications, 1994. **42**(7): p. 2490-2498.

4. Vincent, J.H., *On Some Experiments in Which Two Neighbouring Maintained Oscillatory Circuits Affect a Resonating Circuit.* Proc Royal Society, 1919. **32**(2): p. 84-91.

5. Appleton, E.V., *The Automatic Synchronization of Triode Oscillators.* Proc. Cambridge Phil. Soc.,, 1922. **21**: p. 231-248.

6. deBellescise, H., *La réception Synchrone.* Journal Onde Electronic, 1932.

7. Wendt, K.R. and G.L. Fredendall, *Automatic Frequency and Phase Control of Synchronization in Television Receivers.* Proc. IRE, 1943. **31**.

8. Huntoon, R.D., *Synchronization of Oscillators.* Proc. IRE, 1947.

9. Lacanette, K., *A Basic Introduction to Filters - Active, Passive, and Switched-Capacitor.* National Semiconductor Application Note 779, 1991.

10. Barrett, C. (1999) *Fractional/Integer-N PLL Basics*. Texas Instruments Technical Brief **SWRA029**.

11. Hanumolu, P.K. and M. Brownlee, *Analysis of Charge-Pump Phase Locked Loops.* IEEE trans. on Circuits and Systems, 2004. **51**(9).

12. Abramovitch, D. *Lyapunov Redesign of Classical Digital Phase-Lock Loops*. in *Proceedings of the American Control Conterence Denver*. 2003.

13. O'Keese, W., *An Analysis and Performance Evaluation of a Passive Filter Design Technique for Charge Pump PLL's*, in *National Semiconductor Application Note*. 2001.

14. Abramovitch, D.Y. *Phase-Locked Loops - A Control Centric Tutorial*. in *Proceedings of the American Control Conference*. 2002.

15. Razavi, B., *Monolithic Phase-Locked Loops and Clock Recovery Circuits: Theory and Design*. 1996: Wiley.

16. Samori, C., S. Levantino, and A.L. Lacaita, *Integrated LC oscillators for frequency synthesis in wireless applications.* IEEE Communications Magazine, 2002. **40**(5): p. 166-171.

17. Bianchi, G., *Phase-Locked Loop Synthesizer Simulation*. 2005: McGraw-Hill.

18. Riley, T.A.D., M.A. Copeland, and T.A. Kwasniewski, *Delta-sigma modulation in fractional-N frequency synthesis.* IEEE Journal of Solid-State Circuits, 1993. **28**(5): p. 553-559.

19. Gardner, F.M., *Phaselock Techniques*. 1966: John Wiley and Sons.

20. Staszewski, R.B. and P.T. Balsara, *Phase-Domain All-Digital Phase-Locked Loop.* IEEE trans. on Circuits and Systems, 2005. **52**(3): p. 159-163.

21. Chung, C.-C. and C.-Y. Lee, *An All-Digital Phase-Locked Loop for High-Speed Clock Generation.* IEEE Journal of Solid-State Circuits, 2003. **38**(2): p. 347-351.

22. Chau, Y.A., C.-F. Chen, and K.-D. Tsai. *Design and Analysis of Adaptive-Bandwidth All-Digital Phase-Locked Loop*. in *Proceedings of 2007 International Symposium on Intelligent Signal Processing and Communication Systems*. 2007.

23. Chaivipas, W., P. Oh, and A. Matsuzawa, *All-Digital Phase-Locked Loops, its Advantages and Performance Limitations*, in *International Phd student Workshop*. 2006.

24. Banerjee, D., *PLL Performance Simulation and Design*, in *National Semiconductor* 1998.

25. Curtin, M. and P. O'Brien, *Phase-Locked Loops for High Frequency Receivers and Transmitters - Part 1*, in *Analogue Devices*. 1999.

26. Mirabbasi, S. and K. Martin, *Design of Loop Filter in Phase-Locked Loops.* IEEE Electronic Letters, 1999. **35**(21): p. 1801-1802.

27. Gardner, F.M., *Charge-Pump Phase-Lock Loops.* IEEE trans. on Communications, 1980. **COM-28**(11): p. 1849-1858.

28. Choi, Y., H. Choi, and T. Kwon. *An adaptive bandwidth phase locked loop with locking status indicator*. in *KORUS 2005. Proceedings. The 9th Russian Science and Technology*. 2005.

29. Ge, Y., et al., *A fast locking charge-pump PLL with adaptive bandwidth*, in *ASICON 2005. 6th International Conference On ASIC*. 2005.

30. Carlosena, A. and A. Mánuel-Lázaro, *Design of High-Order Phase-Lock Loops.* IEEE Transactions on Circuits and Systems II: Express Briefs, 2006. **54**(1): p. 9-13.

31. Kamata, M., et al., *Third-order phase-locked loops using dual loops with improved stability*, in *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, 1997*. 1997.

32. Land, R., *Phase and Frequency Stability of PLL systems*, in *Tallinn Technical University Document*. 1992.

33. Carr, J., *RF Components and Circuits*. 2002: Newnes.

34. Stensby, J.L., *Phase Locked Loops Theory and Applications*. 1997: CRC-Press.

35. Kroupa, V.F., *Noise Properties of PLL Systems.* IEEE trans. on Communications, 1982. **30**(10): p. 2244 - 2252

36. Stensby, J.L., *On the PLL Spectral Purity Problem.* IEEE trans. on Circuits and Systems, 1983. **CAS-30**(4).

37. Ascheid, G. and H. Meyr, *Cycle Slips in Phase-Locked Loops: A Tutorial Survey.* IEEE trans. on Communications, 1982. **Com-30**(10): p. 2228-2241.

38. Almeida, T.M. and M.S. Piedade, *High Performance Analog and Digital PLL Design*, in *EEE International Symposium on Circuits and Systems - ISCAS*. 1999.

39. Kroupa, V.F., *Phase Lock Loops and frequency synthesis*. 2003: John Wiley & Sons

40. Thompson, I.V. and P.V. Brennan, *Fourth-order PLL loop filter design technique with invariant natural frequency and phase margin.* IEEE Proc. Circuits Devices Systems, 2005. **152**(2).

41. Norsworthy, S.R., *Delta-Sigma Data Conversions*. 1996: WileyBlackwell.

42. Abramovitch, D.Y., *Lyapunov Redesign of Analog Phase-Lock Loops.* IEEE trans. on Communications, 1990. **38**(12).

43. Williamson, S., *How to Design RF Circuits - Synthesisers*, in *IEE Colloquium on how to Design RF Circuits*. 2000.

44. Hein, J.P. and J.W. Scott, *z-Domain Model for Discrete-Time PLL's.* IEEE trans. on Circuits and Systems, 1988. **35**(11): p. 1393-1400.

45. Lu, J. *Analysis and Design of 5GHz Phase Locked Loops*. in *International Conference on Solid-State and Integrated Circuits*. 2004.

46. Lu, J., et al., *Discrete Z-Domain Analysis of High Order Phase Locked Loops*, in *The 2001 IEEE International Symposium on Circuits and Systems*. 2001.

47. Mansour, M.M. and A. Mehrotra. *Model-Order Reduction Based on PRONY's Method*. in *Design, Automation and Test in Europe Conference and Exhibition, 2003*. 2003.

48. Yap, H.S., *Designing to Digital Wireless Specifications using Circuit Envelope Simulation*, in *Asia-Pacific Micrwave Conference Proceedings*. 1997. p. 173-176.

49. Hedayat, C.D., et al., *High-Level modeling applied to the second-order charge-pump PLL circuit*, in *TI Technical Journal*. 1997.

50. Hedayat, C.D., et al., *Modeling and Characterization of the 3rd Order Charge-Pump PLL: a Fully Event-driven Approach.* Analog Integrated Circuits and Signal Processing, 1999. **19**: p. 25-45.

51. Rantzer, A. *Almost global stability of phase-locked loops*. in *Proceedings of the 40th IEEE conf. on Decision and Control*. 2001.

52. Simon, D. and H. El-Sherief, *Lyapunov Stability Analyses of Digital Phase-Locked Loops*, in *IEEE Conference on Systems, Man and Cybernetics*. 1994.

53. Wu, N.E. *Analog Phaselock Loop Design Using Popov Criterion*. in *Proceedings of American Control Conference*. 2002.

54. Kalman, R.E. and J.E. Bertram, *Control System Analysis and Design Via the Second Method of Lyapunov: (I) Continuous-Time Systems, (II) Discrete-Time systems.* American Society of Mech. Eng., 1959.

55. LaSalle, J. and S. Lafschetz, *Stability by Liapunov's Direct Method*, in *Academic Press*. 1961.

56. de-Smedt, B. and G. Gielen. *Nonlinear Behavioural Modelling and Phase Noise evaluation in Phase Locked Loops*. in *IEEE Proc on Custom Integrated Circuits*. 1998. Santa Clara USA.

57. Popescu, G. and L.B. Goldgeisser. *Mixed signal aspects of behavioral modeling and simulation*. in *Proceedings of the 2004 International Symposium on Circuits and Systems, 2004. ISCAS '04.* . 2004.

58. Arpaia, P., F. Cennamo, and H. Schummy, *Modeling and Characterization of Sigma-Delta Analog-to-Digital Converters",* . IEEE Instrumentation and Measurement Technology, 1998: p. 742 - 752.

59. Malcovati, P., et al., *Behavioral Modeling of Switched-Capacitor Sigma-Delta Modulators.* IEEE trans. on Circuits and Systems, 2003. **50**(3): p. 352-364.

60. Groenendaal, J.G.V.d. and R.M. Braun. *Phase-plane analysis of phase-locked loops used for clock recovery*. in *Proceedings of the 1994 IEEE South African Symposium on Communications and Signal Processing*. 1994.

61. Gundrum, H.C. and M.E. Rizkalla. *Maximizing the stability region for a second order PLL system*. in *Proceedings of the 37th Midwest Symposium on Circuits and Systems*. 1995.

62. Jacobs, O.L.R., *Introduction to Control Theory* 1993: Oxford Press.

63. Mutambara, A.G.O., *Design and Analysis of Control Systems*. 1999: CRC Press.