UNIVERSITY OF CALGARY

# An Agile Frequency Synthesizer for Frequency Hopping Radio

by

Hany H. Shenouda

A THESIS
SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE
DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF
ELECTRICAL AND COMPUTER ENGINEERING

CALGARY, ALBERTA

MARCH, 2000

# Abstract

In this thesis a fast frequency synthesizer is developed. Components of the PLL are presented with linear and transient analysis. The design of the agile synthesizer is detailed including block diagram, schematic implementation, board layout and firmware. The performance of the agile synthesizer is analyzed and compared to the conventional synthesizer. The agile synthesizer shows better dynamic performance over the conventional synthesizer.

# PREFACE

Traditional narrowband communications use fixed channel allocations. In spread spectrum communications a narrow band signal is spread over a large bandwidth.

This investigation looks at a new frequency synthesizer which can be used in frequency hopping applications significantly reducing the time required for retuning the synthesizer when hopping. This not only reduces packet and tuning time overhead but allows faster hopping. The synthesizer is agile and hence called an *agile synthesizer.* With an agile synthesizer hopping rates of 1000 hops per second are possible. Conventional frequency hopping products offer a hopping rate of 40 to 100 hops per second. Commercial products such as the MRX-900 (by Microhard Systems Inc.) and DGR-115 (by Freewave Technologies Inc.) can significantly reduce hopping overhead and increase the hop rate using the agile synthesizer.

Chapter 1 is an introduction to frequency synthesizers. Chapter 2 details the fundamental components of phase lock loops with linear, steady state, and transient analysis. Equations for phase noise are given. A brief introduction to Fractional-N Synthesizer is made. Finally, a brief note about frequency chirps is given at the end of the chapter.

Chapter 3 details the design of the agile synthesizer and includes a Matlab simulation comparing the performance of coherent and non-coherent frequency hopping. Schematics are given in chapter three along with the gerber plots for the board layout. Firmware that is developed is found in Appendix B.

Chapter 4 discusses the performance of the agile synthesizer. Spectral plots and dwell time plots are shown. Static and dynamic performance are presented and a comparison is made between the conventional synthesizer and the agile synthesizer. Chapter 5 concludes the thesis with a summary of results.

# ACKNOWLEGDMENTS

I would like to thank my supervisor, Dr. Michel Fattouche for his help and guidance. I would like to thank Microhard Systems Inc. for the use of their laboratory facility and their CAD software.

I would like to thank Dr. John McRory. He has been an excellent advisor and teacher and has ignited my interest in the RF field.

*Dedicated to my parents for their support and Barbara for her encouragement.*

# CONTENTS

# LIST OF FIGURES

# LIST OF TABLES



# LIST OF SYMBOLS

| | |
|---|---|
| $K_{vco}$ | DC gain constant of the VCO |
| $K_{\varnothing}$ | Phase comparator gain constant |
| $f_{VCO}$ | VCO output frequency |
| $f_{ref}$ | Reference frequency |
| $f_o$ | Center frequency |
| N | Reference Divider |
| G(s) | Loop Filter Transfer Function |
| $\eta$ | Input noise spectral density |
| $\omega_c$ | The loop filter cutoff frequency |
| $B_L$ | Phase Lock Loop 3dB Bandwidth |
| $\omega_c$ | The loop filter cutoff frequency |
| $\Omega$ | Measure of resistance ohms |
| mA | Measure of current milliamperes |
| m(t) | Message signal |
| s(t) | Transmitted signal |
| I | Inphase component |
| Q | Quadrature component |

# LIST OF ABBREVIATIONS

| | |
|---|---|
| AWGN | Additive White Gaussian Noise |
| FLT | Filter |
| VCO | Voltage Controlled Oscillator |
| BPF | Bandpass Filter |
| PA | Power Amplifier |
| T/R | Transmit/Receive |
| μP | Microprocessor |
| FSK | Frequency Shift Keying |
| RF | Radio Frequency |
| IF | Intermediate Frequency |
| PLL | Phase Lock Loop |
| LO | Local Oscillator |
| SSB | Single Side Band |
| IC | Integrated Circuit |
| FET | Field Effect Transistor |
| FM | Frequency Modulation |
| AM | Amplitude Modulation |
| QAM | Quadrature Amplitude Modulation |
| QPSK | Quadrature Phase Shift Keying |
| PSK | Phase Shift Keying |
| DSO | Digital Sampling Oscilloscope |
| BER | Bit Error Rate |

# CHAPTER ONE

## 1. Introduction

Spread Spectrum communications has seen significant commercialization since its introduction by the military in the early 1950's. There are at present three types of commercial spread spectrum systems: Direct Sequence spread spectrum, Frequency Hopping spread spectrum and a hybrid system using both Frequency Hopping and Direct Sequence. Another Spread Spectrum technique, which is mainly restricted to military and radar applications, is Frequency Chirp.

## 1.1 Frequency Hopping

In a frequency hopping system the available channel bandwidth is subdivided into distinct channels. The carrier frequency is then changed such that the frequency-time allocation scheme is pseudo-random.

For an AWGN channel PSK gives a better performance than FSK, however, it has been traditionally difficult to maintain phase coherence across wideband hops [24]. Consequently, phase modulations schemes and coherent frequency modulation are not widely used in frequency hopping systems. Frequency hopping systems are traditionally slow. This means that the carrier hopping rate is much less than the bit rate.

Currently, frequency hopping systems carry significant overhead when re-synthesizing the new hopping channel. The objective of this research project is to introduce a new agile synthesizer that significantly reduces the overhead for synthesizing the hopping channel.

## 1.2 Typical Transceiver Design

Digital frequency synthesis has revolutionized modern radio design. It allows multiple frequencies to be accurately generated using a single oscillator reference and a voltage controlled oscillator. Spread spectrum systems using frequency hopping rely on a microcontroller/microprocessor to program the pseudo-random hopped channels.

A typical transceiver is shown in Figure 1.2.1. At the heart of the radio is the Phase Lock Loop (PLL) which is used to generate $LO_1$ for the transmitter and for the receiver. Input data modulates $LO_1$ which is then amplified by the Power Amplifier (PA). The front end switch selects between the transmit and receive (T/R) paths.

In the receiver, the signal is first amplified by the LNA (Low Noise Amplifier) and filtered through the Bandpass Filter (BPF). Then the signal is down converted to the 1$^{st}$ IF where it is filtered. A second mixer converts 1$^{st}$ IF signal to a lower frequency the 2$^{nd}$ IF. A Limiter is used to limit the signal amplitude removing AM modulation. The signal is then demodulated using a quadrature demodulator.

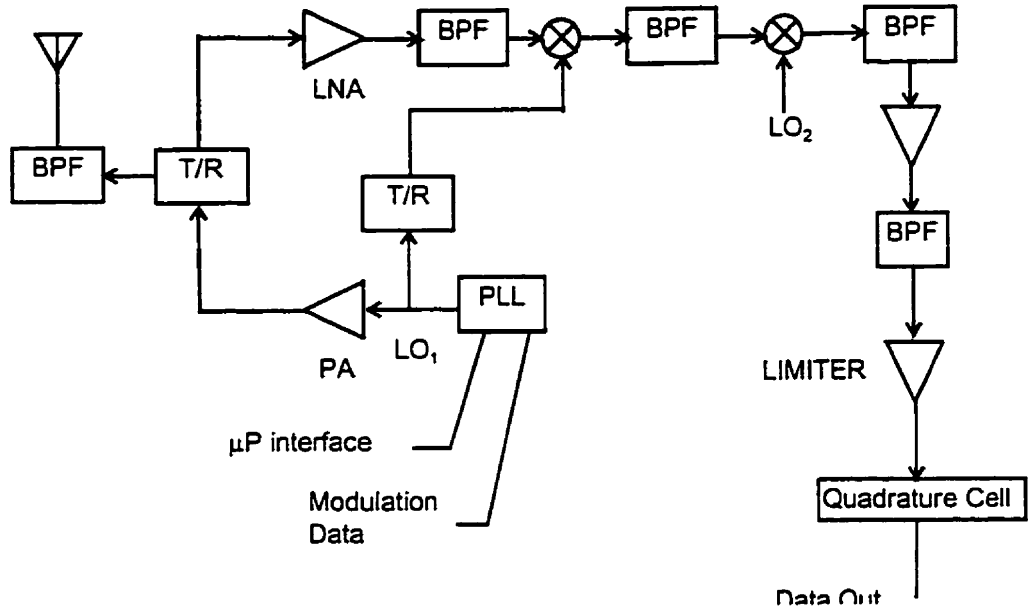



Fig. 1.2.1 Typical Radio Block Diagram

Only one PLL is required since this transceiver implements a Time Division Duplex scheme. The Phase Lock Loop allows a digital control unit to program the frequency to be synthesized. The LO synthesized frequencies for transmit and receive are different. Every time the radio is switched from transmit to receive or vise versa the LO must be reprogrammed and the radio must wait to lock to the new frequency.

## 1.2.1 Phase Lock Loops

Phase-locked loops are commonly used in electronics. In a modern radio design there is at least one PLL in a radio. PLLs are circuits in which the phase of two signals are compared and an attempt is made to lock them together.

For instance, in a demodulator the incoming signal is phase locked with a local oscillator and the information is extracted as the error between the two. Phase-lock loops look deceivingly simple. There are four fundamental components to a PLL as shown in Fig 1.2.2 [1].

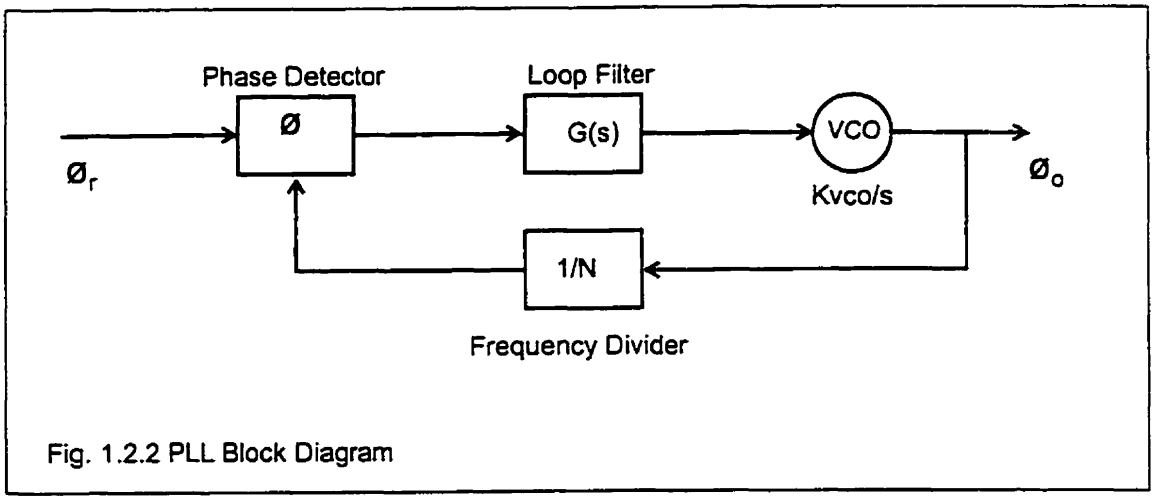



Fig. 1.2.2 PLL Block Diagram

The first component is a VCO which provides a voltage to frequency transformation. Next is the frequency divider which generates a comparison frequency. The phase comparator gives the phase error between the reference phase and the feedback phase. Within a given range the phase comparator is assumed to be linear. This assumption will simplify the analysis of the PLL and is accurate for steady state analysis. The final component is the loop filter which affects the transient and the steady state performance of the PLL.

### 1.2.2 Acquisition Time

The frequency synthesizer is a control system. The response of the PLL in switching from one frequency to another takes a finite amount of time known as the acquisition time or the lock time. It can be modeled as a frequency step response to the PLL. During acquisition the synthesizer cannot be used to transmit or to receive. Instead, the entire radio must wait until the synthesizer is stable before it can be used. Acquisition creates an overhead in frequency hopping systems and in TDMA systems where the LO must be retuned frequently.

Acquisition time can be reduced by trading-off steady state performance. For instance, decreasing the lock time by decreasing the feedback divider (or increasing the reference frequency). This will decreases the resolution of the synthesizer.

### 1.2.3 Fast Frequency Hopping

Fast hopping refers to a scheme where one bit is spread over more than one hopped channel. This method can be implemented using a coherent or a noncoherent receiver. Slow hopping has many bits sent on the same channel. The advantages of fast hopping with a coherent receiver implementation are well documented in [24].

Fast frequency hopping is not achievable with the scheme researched here. However, an agile synthesizer will allow much faster hopping times. This will increase immunity to large interference and will reduce overhead.

## 1.3 Thesis Outline

This research involves a computer simulation of the agile synthesizer, as well as design of the agile synthesizer and lab measurements. The main focus of the research is the transmitter. However, in this thesis a receiver structure is included for completeness. Possible future research work could focus on a receiver implementation.

Chapter 2 discusses the Phase Lock Loop linear behavior and transient response. A discussion of the RF solid state switches is also covered in Chapter 2.

Chapter 3 illustrates the design of the agile synthesizer. It will include the design of the synthesizer and the design of a switch matrix. Furthermore, the test setup is shown in chapter 3.

Chapter 4 shows the performance of the agile synthesizer. It shows the steady state and transient behavior. Chapter 4 also compares the phase noise performance of the agile synthesizer to the conventional synthesizer. It discusses the spurious noise in the agile synthesizer.

Chapter 5 concludes by summarizing the results obtained and giving recommendations for future research.

# CHAPTER TWO

## 2. Agile Frequency Synthesis

An agile frequency synthesizer can switch frequency much faster than an ordinary PLL synthesizer. Although the agile synthesizer has a minimum dwell time, the transition from one frequency to another is rapid. The dwell time is the interval in which the synthesizer sits on a particular frequency.

## 2.1 Fundamental Components of Digital Phase Lock Loops

The linear model of the PLL is shown in Fig. 2.1.1. The model shows the four distinct components to the PLL. The phase detector, loop filter, VCO, and the frequency divider. The next sections explain each component in more detail.

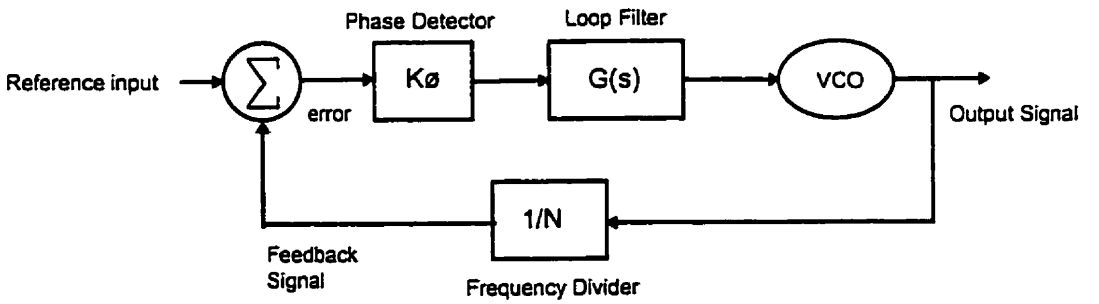



Fig. 2.1.1 Basic Phase Lock Loop Structure

## 2.1.1 Phase Detector

The phase detector gives a DC output voltage proportional to the phase difference between the input signals. The two types of phase detectors are multiplier phase detectors and a sequential phase detectors [1].

Multiplying phase detectors are simply analogue multipliers to which the output port is DC coupled. These types of detectors are reserved for applications where low SNR conditions exist. Consequently, multiplying type detectors are used in demodulating phase lock loops.

Sequential phase detectors function as digital gates and flip-flops by relying on the relative timing of edges. Although, this scheme does not work well for SNR less than 10dB, it offers superior performance capture and tracking [1].
Fig 2.1.2 shows a simple sequential phase detector implementation.

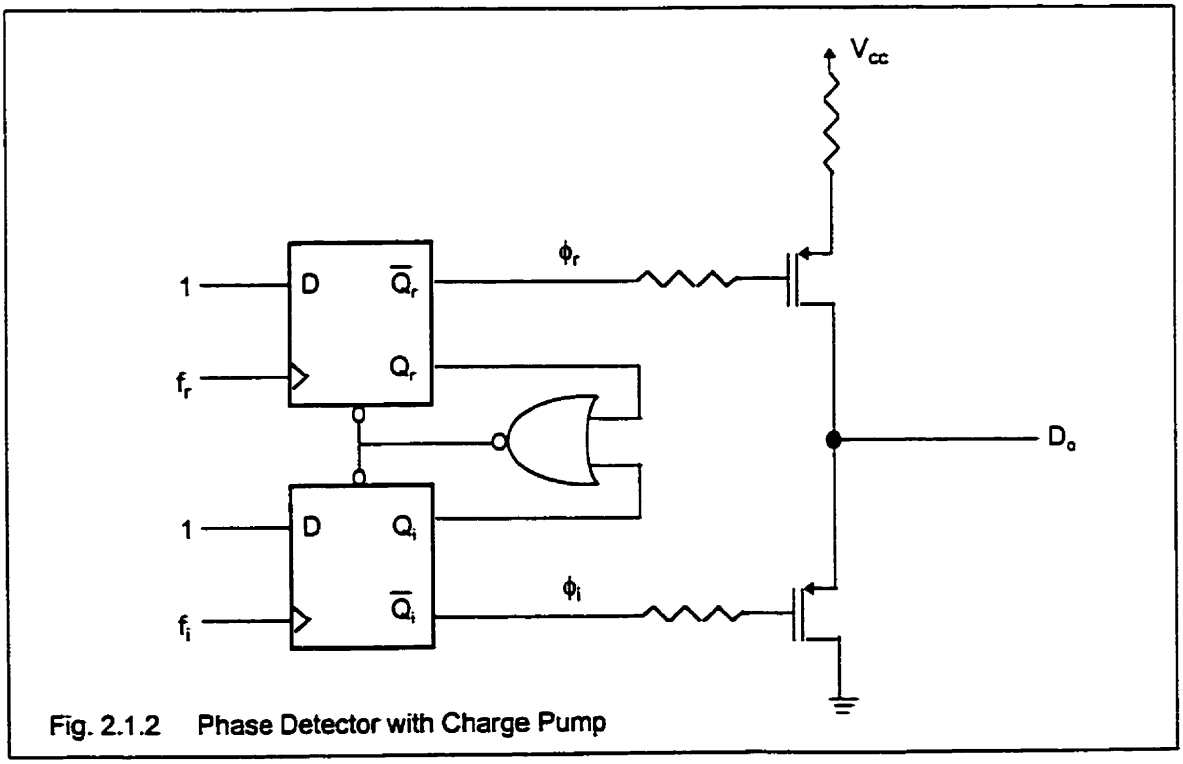



Fig. 2.1.2 Phase Detector with Charge Pump

The input signals $f_i$ and $f_r$ drive clocks for separate flip-flops. The flip-flops are cleared by an XNOR gate which occurs only when $f_i$ and $f_r$ are both high or both low. Thus if $f_i$ and $f_r$ are locked such that there is no difference in the relative edges of $f_i$ and $f_r$ the flip-flops are also cleared. The flip-flops circuitry drives a CMOS Switch which acts as a charge pump. When both flip-flops are cleared the charge pump is high impedance.

Sequential phase detectors are able to discriminate both phase and frequency and hence do not lock onto odd harmonics as does the multiplying phase detector. In frequency synthesis sequential phase detector PLLs are used almost exclusively. Low SNR is not a problem in synthesizer applications since the VCO output is always sufficiently above the noise floor. The sequential phase detector has a very large capture range and thus is able to pull in and lock the VCO from a very large frequency offset. Furthermore, the sequential phase detector does not lock onto harmonics. Odd harmonics (third harmonic) are usually only -30 dBc at the output of a VCO. Sequential phase detectors are used in nearly all frequency synthesis PLL.

A closer look at sequential phase detectors shows that there is a dead-zone [14], in which the phase detector shows non-linear gain variation with VCO control voltage and temperature. A PLL operates in the dead zone when the phase error approaches zero or the PLL is locked to the proper frequency. Non-linear operation is caused by the finite time the current sources take to switch on when the phase detector input signal edges are very close together. However, by introducing delays in the phase detector reset path, the dead-zone problems are reduced by giving the current sources more time to switch on.

## 2.1.2 Voltage Controlled Oscillator

A VCO is an oscillator whose frequency is controlled by an applied voltage to the tuning port. The frequency of oscillation in a VCO is controlled by a varactor diode. This device exhibits different capacitance depending on the applied reverse bias voltage. Circuits with a Q of 40 to 400 can be built by using a varactor in conjunction with a microstrip line, inductor, or resonator. The circuit's resonating frequency is then adjusted by varying the applied voltage to the varactor.

The design of the VCO is not under considered, instead, only a simple model is derived. In this investigation all VCOs are commercially available along with detailed specifications.

An oscillator can be modeled as an amplifier in a positive feedback loop.

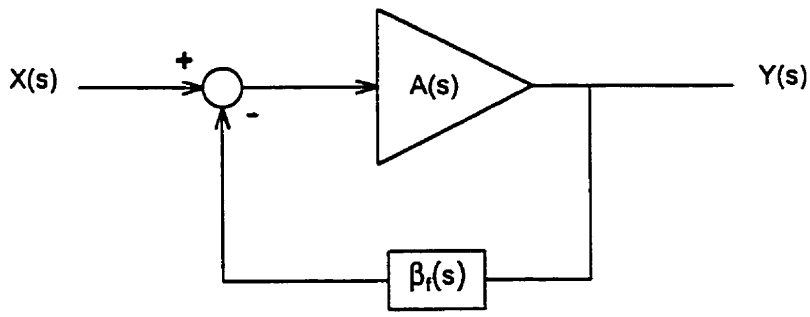



Fig. 2.1.3 Oscillator Model

The transfer function for the amplifier is:

$$A_f(s) = \frac{A(s)}{1 - \beta_f(s)A(s)} \qquad (2.1.2.1)$$

where,

$A_f$ (s) = the complete transfer function of the amplifier with feedback

A(s) = the transfer function of the amplifier

$\beta_f$ (s) = the feedback component

According to the Barkhausen criterion, when $\beta_f(s)A_f(s) = 1$, the feedback amplifier becomes an oscillator. Under these conditions the imaginary component is zero [13]. Noise starts the oscillation and the amplitude of the signal grows until the amplifier becomes saturated. The frequency of oscillation makes $\beta_f(s)A_f(s) = 1$. There may be multiple frequencies at which the Barkhausen criterion is satisfied which will result in multi-frequency oscillations occurring.

Phase noise in oscillators is a critical parameter. The phase noise may set the limits for dynamic range and sensitivity in a wireless system. An ideal oscillator output is a single tone with no phase noise:

$$V(t) = V_o \sin(2\pi f_o t + \theta) \qquad (2.1.2.2)$$

where $V_o$ is the nominal voltage of the oscillator and $f_0$ is the nominal frequency of the oscillator. For a real oscillator, the equation is:

$$V(t) = (V_o + E(t)) \sin(2\pi f_o t + \theta + \phi(t)) \qquad (2.1.2.3)$$

where $E(t)$ is the magnitude of the random voltage variations and $\phi(t)$ is the random phase variations.

The spectral output does not distinguish between noise from E(t) and noise from $\phi(t)$; however, since oscillators are operating in saturation the limiting action makes the AM component due to E(t) insignificant when compared to $\phi(t)$. The AM noise which is caused by E(t) is about 20dB less than the FM noise caused by $\phi(t)$ [13].

To design the loop filter the VCO tuning characteristic must be known. The VCO tuning curves are non-linear since the varactor's voltage versus capacitance characteristic is non-linear. An average value of the gain constant over the frequency range of interest is used and denoted by $K_{vco}$ where:

$$K_{vco} = \frac{\partial \omega_o}{\partial V_t} \qquad (2.1.2.4)$$

where $\omega_o$ is the output frequency and $V_t$ is the tuning voltage.

### 2.1.3 Loop Filter

The loop filter is used to suppress sidebands which occur at the reference frequency. Reference sidebands arise due to a small proportion of the AC ripple from the phase detector passing through the loop filter and phase modulating the VCO [1].

Although any kind of spurious output is not desired, reference sidebands are especially a problem since they occur at the reference frequency. The reference frequency is chosen equal to the channel spacing for optimum loop performance. Hence, reference sidebands appearing in the LO decreases adjacent channel rejection by mixing in the adjacent channels during down conversion. For a transmission, reference sidebands in the LO radiate energy into the adjacent channels which increases the problem of the Near and Far Effect.

Any one who has designed a synthesizer knows how important the loop filter is in determining the parameters of the synthesizer. Loop filters can be active or passive. For this investigation the scope is limited to passive filters which are less noisy.

The design of loop filters requires the use of a bode plot. The loop filter must be designed such that the closed loop system (the whole synthesizer) is stable.

Loop filters are classified based on order and type. The order of the PLL is always one higher than the order of the loop filter due to the integrating action of the VCO.

The loop filter and reference frequency determine critical parameters such as phase jitter, reference side band suppression and lock time.

### 2.1.4 Frequency Divider

The frequency divider is the section where the VCO frequency ($f_{vco}$) is divided down to the reference frequency and then used for comparison with $f_{ref}$. The division ratio (N) is an integer value for conventional phase lock loops. Section 2.4 discusses the use of non-integer divide ratio in a class of PLL synthesizers called Fractional N-Synthesizers.

The frequency divider can be divided into two sections. First, is a Prescalar which is usually made from ECL technology and is capable of operation into the GHz range. Second, is a programmable counter made of CMOS technology which can be used in the 100MHz range. The duty cycle of the frequency divider should be as close to 50% as possible.

An improvement over a fixed divider scheme is to use *dual-modulus prescaling* [1]. This makes use of a high frequency divider which divides by a programmable value P or P+1 depending on the modulus control input. A special low frequency counter is used to control the division ratio of the prescalar through the modulus control input [1]. The major advantage of the dual modulus prescalar is that it allows the output frequency to be synthesized in integer multiples of the reference frequency and also reduces phase noise [15].

The net result of the dual-modulus prescaling is that the input frequency of the VCO is divided down by P+1 for A times and P for N-A times [1]. The effective division ratio is $N_t$ where,

$$N_t = NP + A \qquad (2.1.4.1)$$

where $N \geq A$

## 2.2 Analysis of Phase Lock Loops

Basic analysis of phase lock loops can be done using control system principles. Transient analysis is however more difficult since non-linear behavior may occur. This section will first describe the linear behavior which is necessary for closed loop stability, second it will look at noise performance, and third the transient response of the phase lock loop. The agile synthesizer's performance is dependent on the phase lock loop filter's steady state and transient characteristics.

### 2.2.1 Linear Analysis

Figure 2.2.1. represents the linear model for the PLL that can be used for analysis. The VCO contributes a pole and a gain term Kvco acting as an integrator with a gain constant. The loop filter is represented by the transfer function G(s) in the Laplace domain. Ignoring the effect of the dead zone the phase detector can be modeled as a gain constant Kø. The phase error $\Theta_e$ is derived by comparing the reference phase $\Theta_r$ to feedback term $\Theta_i$. The frequency divider generates the feedback phase $\Theta_i$ from the output frequency. The term $\Theta_o$ is the phase of the synthesized frequency.

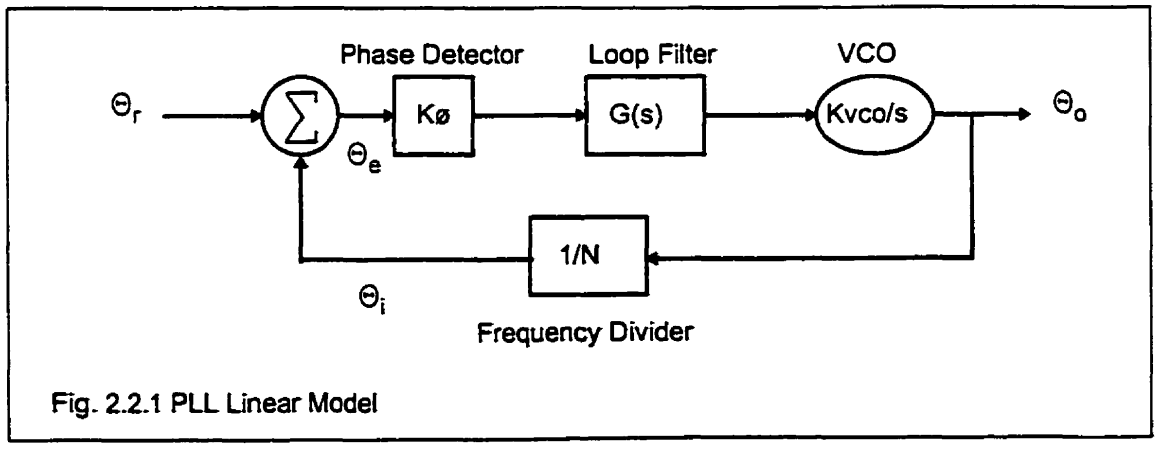



Fig. 2.2.1 PLL Linear Model

Using basic control systems theory the closed loop transfer function is given by:

$$\frac{\Theta_o}{\Theta_r} = \frac{NK_\phi K_{vco} G(s)}{sN + K_\phi K_{vco} G(s)} \qquad (2.2.1.1)$$

The loop filter with a transfer function, G(s), can be designed as an active or passive lowpass filter. We will concentrate on the passive low pass filter for this investigation. Figure 2.2.2 shows a general structure for a passive third order lowpass filter. The transfer function is defined from the charge pump current output $I_p(s)$ to the VCO tuning voltage $V_o(s)$:

$$G(s) = \frac{V_o(s)}{I_p(s)} \qquad (2.2.1.2)$$

Using linear circuit theory, the loop filter transfer function is given by:

$$G(s) = \frac{sC_2R_2 + 1}{[s^2C_1C_2R_2 + sC_1 + sC_2\,](\,sR_3C_3 + 1)} \qquad (2.2.1.3)$$

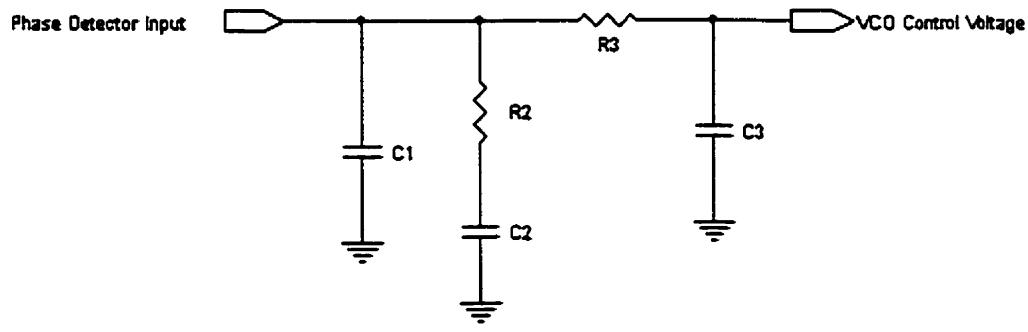



Fig. 2.2.2 Third order Lowpass Loop Filter

The loop filter determines critical parameters of the PLL synthesizer such as lock time, reference sideband suppression, gain margin and phase margin. The PLL must be closed loop stable and should practically have a gain margin of more than 15dB and a phase margin of more than 30°.

The open loop gain from $\Theta_e/\Theta_i$ is given by:

$$\frac{\Theta_e}{\Theta_i} = \frac{K_\phi G(s) K_{vco}}{sN} \tag{2.2.1.4}$$

From the open loop analysis general design equations can be derived for the third order lowpass filter in Fig 2.2.2. These equations are summarized below for convenience [3].

$$T_1 = \frac{\sec\phi_p - \tan\phi_p}{2\pi f_p} \tag{2.2.1.5}$$

$$T_3 = \frac{\sqrt{10^{\alpha/20} - 1}}{2\pi f_{ref}} \tag{2.2.1.6}$$

$$f_c = \frac{1}{2\pi}\frac{\tan\phi(T_1+T_3)}{(T_1+T_3)^2 + T_1T_3}\left[\sqrt{1+\frac{(T_1+T_3)^2+T_1T_3}{[\tan\phi(T_1+T_3)]^2}} - 1\right] \tag{2.2.1.7}$$

$$T_2 = \frac{1}{(2\pi f_c)^2(T_1+T_3)} \tag{2.2.1.8}$$

$$\omega_C = 2\pi f_c \tag{2.2.1.9}$$

$$C_1 = \frac{T_1 K_\phi K_{vco}}{T_2 \omega_c^2 N}\sqrt{\frac{(1+\omega_c^2 T_2^2)}{(1+\omega_c^2 T_1^2)(1+\omega_c^2 T_3^2)}} \tag{2.2.1.10}$$

$$C_2 = C_1\left(\frac{T_2}{T_1} - 1\right) \tag{2.2.1.11}$$

$$R_2 = \frac{T_2}{C_2} \qquad (2.2.1.12)$$

where,

$\alpha$ = the additional attenuation over the second order lowpass filter in dB

$K_\phi$ = the phase detector/charge pump current

$f_p$ = the gain crossover frequency of the open loop PLL

$\phi_p$ = the phase crossover frequency for the open loop PLL

$\omega_c$ = the loop filter cutoff frequency

$f_{ref}$ = the loop reference frequency

$T_1, T_2, T_3$ = time constants of the loop filter

The value of $R_3$ should be between 10kΩ to 20kΩ, then $C_3$ can be calculated.

$$C_3 = \frac{T_3}{R_3} \qquad (2.2.1.13)$$

### 2.2.2 Noise Performance for PLL

No discussion of the PLL is complete without a fundamental noise analysis. Additive noise at the input of a PLL cannot simply be translated into additive noise at the output of the VCO. The VCO may only be phase or frequency modulated. Instead, the low frequency noise components are superimposed on the DC signals inside the loop filter which results in phase noise on the VCO output or *phase jitter* [1]. Amplitude modulation of the VCO is insignificant due to the limiting action of the oscillator [4].

The following equation represents the output of the VCO where the noise has been phase modulated through the loop filter and the VCO:

$$V_i(t) = A_c \cos(\omega_c t) + A_n(t)\cos(\omega_c t + \theta_n(t)) \qquad (2.2.2.1)$$

Using the trigonometric identities to resolve the in-phase and quadrature components:

$$V_i(t) = \left\{A_c + A_n(t)\cos(\theta_n(t))\right\}\cos(\omega_c t) - A_n(t)\sin(\theta_n(t))\sin(\omega_c t) \qquad (2.2.2.2)$$

It is assumed that the noise is uniformly distributed within the input bandwidth $B_i$. The noise input power spectral density is:

$$\eta \equiv \frac{A_n^2}{2Bi} \qquad (2.2.2.3)$$

The limiting action removes the in-phase component leaving the quadrature noise component. The power of the quadrature component is:

$$A_{nq}^2 = \frac{A_n^2}{2} \tag{2.2.2.4}$$

The mean square value of the effective phase jitter on the input signal $\overline{\Delta\phi_i^2}$ is given by:

$$\overline{\Delta\phi_i^2} = \overline{\arctan^2\left(\frac{A_{nq}}{A_c}\right)} \tag{2.2.2.5}$$

For small angle approximations the mean square value of the effective phase jitter on the input signal becomes:

$$\overline{\Delta\phi_i^2} = \left(\frac{\overline{A_n^2}}{2A_c^2}\right) \tag{2.2.2.6}$$

In terms of SNR the phase jitter is simply:

$$\overline{\Delta\phi_i^2} = \frac{1}{2SNR_i} \tag{2.2.2.7}$$

Assuming that the loop bandwidth $B_L < B_i / 2$, all components are passed through the loop, then the means square jitter on the VCO output is [1]:

$$\overline{\Delta\phi_o^2} = \frac{B_L N^2}{SNR_i B_i} \tag{2.2.2.8}$$

The result can be simplified as follows:

$$\sqrt{\overline{\Delta\phi_o^2}} = N\sqrt{\frac{B_L}{C/n_o}} \qquad (2.2.2.9)$$

$$N = \frac{f_c}{f_{ref}} \qquad (2.2.2.10)$$

where,

$\sqrt{\overline{\Delta\phi_o^2}}$ = the rms phase jitter at the output of the VCO in radians.

$C/n_o$ = the input carrier to noise spectral density ratio

Measurements on PLL noise performance do not completely agree with the equation 2.2.2.9. By including a residual phase jitter term more accurate results are obtained.

$$\sqrt{\overline{\Delta\phi_o^2}} = \sqrt{N^2\frac{B_L}{C/n_o} + \overline{\Delta\phi_v^2}} \qquad (2.2.2.11)$$

where,

$\overline{\Delta\phi_v^2}$ = the square mean residual phase jitter

The residual phase jitter is due to oscillator noise which is discussed in more detail by Randall Rhea [6]. From the noise discussion it is clear that the comparison frequency is strongly related to the phase jitter. The higher the comparison frequency, the smaller the divide ratio N, the smaller the rms phase jitter. However, large comparison frequencies do not allow small channel spacing. The agility of the synthesizer is also improved by increasing the comparison frequency.

### 2.2.3 Transient Response

Linear methods can be used to approximate phase and frequency transients provided that the frequency step is small such that the phase detector still operates inside its linear limits. In this section a second order type II loop with unity damping is used. A similar analysis can be derived for third order filters see [1] for more information.

The time domain output phase response of the loop is given by [1]:

$$\Theta_o(t) = N\Theta_i(t)(1-e^{-\omega_n t}) \qquad (2.2.3.1)$$

where

$\omega_n$ = the natural frequency of the loop

The phase error is the difference between the feedback phase and the input reference phase [1].

$$\Theta_e(t) = \Delta\omega t e^{-\omega_n t} \qquad (2.2.3.2)$$

where,

$\Delta\omega$ = the initial frequency error.

By differentiating the phase response of the loop with respect to time the frequency error response is obtained [1].

$$\omega_e(t) = \Delta\omega(1-\omega_n t)e^{-\omega_n t} \qquad (2.2.3.3)$$

The maximum output step for which the phase detector remains in its linear range of operation given by [1]:

$$\Delta f_o(\max) = 2\pi eNfn \tag{2.2.3.4}$$

The transient response can be extended to the loop filter given in Fig. 2.2.2 by simply applying the transfer function:

$$\Theta_o(s) = \frac{\Delta\omega}{s^2}\frac{NK_\phi K_{vco}G(s)}{[sN + K_\phi K_{vco}G(s)]} \tag{2.2.3.5}$$

Substituting for G(s),

$$\Theta_o(s) = \frac{\Delta\omega NK_\phi K_{vco}}{s^2}\frac{sC_2R_2 + 1}{s^2N(sC_1C_2R_2 + C_1 + C_2)(sC_3R_3 + 1) + K_\phi K_{vco}(sC_2R_2 + 1)} \tag{2.2.3.6}$$

And taking the inverse LaPlace transform to find the output phase response to the filter. The impulse response and the step response are performed easily in Matlab. However, it is important to remember that the transfer function is valid only over the linear range of the phase detector.

## 2.3 Fractional-N Synthesis

Until this point the discussion of phase lock loops referred to an integer division ratio N. Since the division ratio N sets the resolution of the synthesizer, it cannot be made arbitrarily small. This caps the maximum loop reference frequency limiting its transient response and setting time. For example, in AMPS where the channel spacing is 30KHz the maximum loop reference frequency is 30KHz. In frequency hopping systems where rapid hopping and tight channel spacing are required, a significant penalty is incurred with conventional synthesizers.

Fractional-N synthesizers operate by effectively using a fractional value for N. This allows high synthesizer resolution while still maintaining a high loop reference frequency. Fractional-N synthesizers operate by quickly alternating between values N and N+1. The resolution of the output frequency is $f_{ref}$ / q. The synthesizer output given below [1]:

$$f_c = \left[ N + \frac{p}{q} \right] f_{ref} \qquad (2.3.1)$$

where,

$1 \leq p \leq q-1$

$f_c$ = the output carrier frequency

q = the accummulator register

However, there are drawbacks to using a Fractional-N synthesizer. The alternating action of the divisor phase modulates the carrier producing sidebands from $f_{ref}/q$ to $f_{ref}$. Techniques to suppress sidebands including open loop compensation to reduce the spurious output [1]. In this investigation, the agile synthesizer designed uses integer-N synthesizers, however, it may be extended to utilize Fractional-N synthesizers in the future.

## 2.4 Solid State RF Switches

The two families of RF switches are RF solid state switches and RF relay switches. This section will focus on solid state RF switches. An ideal switch has a zero turn on time, zero insertion loss and infinite isolation. Real switches have frequency dependent responses.

### 2.4.1 GaAs Switch vs PIN Switch

The two main types of solid state switches are PIN and GaAs switches. The PIN diode switch is like a current controlled resistor at frequencies well above the diode cutoff frequency [16]

$$f_{cutoff} = \frac{1}{2\pi\tau} \qquad (2.4.1.1)$$

where,

$\tau$ = the minority carrier lifetime of the device

When the PIN diode is operated at a frequency of $10f_{cutoff}$ the resistive behavior of the diode is frequency independent. This makes the PIN diode suitable for wide band attenuator and switch applications. However, its usable lower frequency range is limited [16].

Fig. 2.4.1 shows a typical anti-series configuration for a PIN diode switch [17]. The anti-series configuration provides a low distortion switch with excellent wide band performance. The switch is current controlled with on-resistance of less than 1Ω for 10mA bias [17].

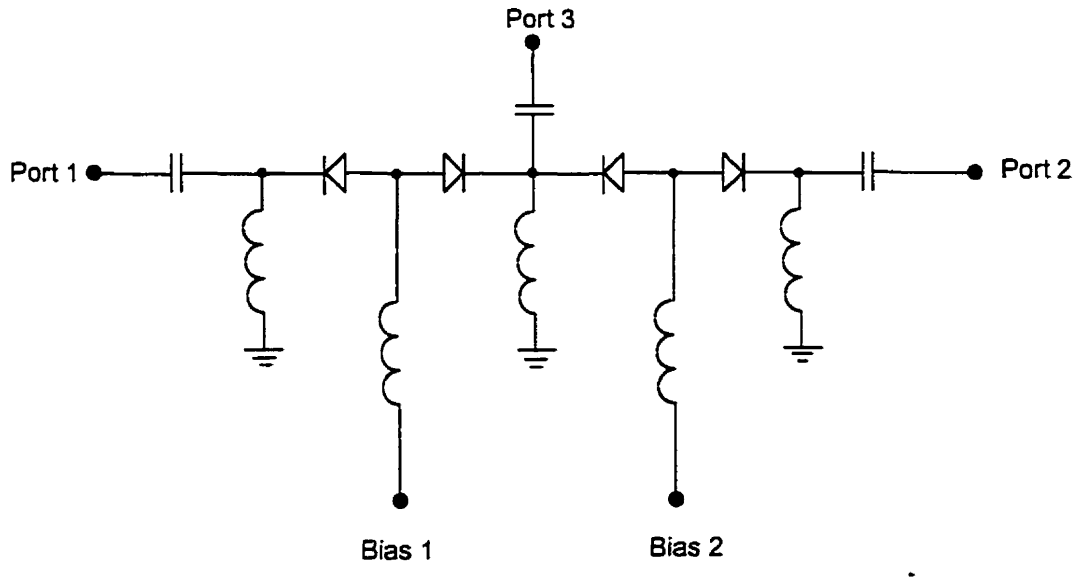



Figure 2.4.1 PIN Diode Switch with Control Lines

When implementing this circuit decoupling capacitors should be used on the control lines. Inductors should be chosen with a high reactance so as to achieve good wide band performance. Table 2.5.1 shows the logic configuration for coupled ports.

Table 2.4.1 Switch Control Logic

| **Coupled Ports** | **Bias 1** | **Bias 2** |
|---|---|---|
| Port 1 to Port 3 | High | Low |
| Port 2 to Port 3 | Low | High |

The other type of switch is a GaAs switch. These consist of a FET on a GaAs substrate. FET are voltage controlled devices, unlike PIN diodes they do not require high bias currents. GaAs FET switches have a broader frequency range than PIN diode switches since they can pass low frequency components with good isolation through the FET channel. The switching times for the GaAs switches is in the order of tens of nanoseconds; where as for PIN diodes, it is in the 100's of nanoseconds. Furthermore, video leakage, the switching-signal leakage into the RF ports, is 10dB less in GaAs FET than in PIN switches [18]. However, PIN diode switches can handle more power as the limit is determined by the average power dissipation whereas the GaAs switches are limited by the FET internal voltage and current ratings [18].

### 2.4.2 Switch Performance

Insertion loss is the attenuation between the active input and output ports of the switch. For GaAs FET switches the insertion loss is less than 1dB over the specified frequency range. The noise figure of the switch is equal to its insertion loss. Isolation is the attenuation between the inactive input and output ports. Typically the isolation of a switch will range from 20-30 dB. More isolation can be achieved by cascading multiple switches and by reducing RF coupling on power and control lines. Switches act like mixers and will couple switching transients to the RF ports. These transients can be reduced by slowing the transition rate between active and inactive states.

There are reflective and absorptive switches. Reflective switches have a high VSWR at the inactive port. Absorptive switches have a low VSWR at the inactive port.

## 2.5 Rapid Frequency Hopping vs Frequency Chirps

The discussion of the agile synthesizer is incomplete without some mention of frequency chirping. The spread spectrum method of frequency chirping has emerged from military radar applications. Frequency chirp is different from rapid frequency hopping in two fundamental ways. First, frequency chirps are continuous phase spreading techniques, whereas hopping involves discrete channels or frequency steps. Second, frequency chirps can ramp in frequency at a much faster rate than possible by rapid frequency hopping.

Rapid hopping has a minimum dwell time limited by the tuning time. Although the dwell time in a rapid hopping scheme can be reduced to less than 1 ms, the frequency time functions are at best only staircase approximation of a frequency chirp. Frequency chirps are described mathematically [12]:

$$s_r(t) = m(t)e^{j2\pi(f_o t + kt^2/2)} \tag{2.5.1}$$

where s(t) is the transmitted signal and m(t) is the information signal. The variable k represents the frequency acceleration term in units of rad/s$^2$. Applying the correlation function [12]:

$$\chi_u(\tau,\upsilon) = \int u(x)u^*(x+\tau)e^{j2\pi ux}dx \tag{2.5.2}$$

to the frequency chirp, we obtain:

$$\chi_u(\tau,\upsilon) = \int_{-\infty}^{\infty} m(t)m^*(t+\tau)e^{-j2\pi(k\tau-\upsilon)t}e^{-j2\pi(f_o+\frac{k\tau}{2})\tau}dt \tag{2.5.3}$$

Assuming the message signal m(t) consists of the a time limited square pulse m(t) where:

$$m(t) = \frac{1}{\sqrt{\tau}} rect(\frac{t}{\tau'}) \tag{2.5.4}$$

Then by substituting m(t) in to the autocorrelation function

$$\chi_u(\tau,\upsilon) = \frac{e^{-j2\pi k\tau^2}}{\tau'} \int_{-\infty}^{\infty} rect(\frac{t}{\tau'}) rect(\frac{t+\tau}{\tau'}) e^{-j2\pi(k\tau-\upsilon)t} dt \tag{2.5.5}$$

The integrand is zero outside $-\tau'/2 < t < (\tau'/2) - \tau$. For all $\tau$

$$\chi_u(\tau,\upsilon) = e^{j\pi\upsilon\tau} \frac{\sin[\pi(k\tau-\upsilon)(\tau'-|\tau|)]}{\pi(k\tau-\upsilon)\tau'} rect(\frac{\tau}{2\tau'}) \tag{2.5.6}$$

The time autocorrelation function is given by setting $\upsilon = 0$ [12].

$$\chi_u(\tau,0) = e^{j\pi\upsilon\tau} \frac{\sin[\pi k\tau(\tau'-|\tau|)]}{\pi k\tau\tau'} rect(\frac{\tau}{2\tau'}) \tag{2.5.7}$$

In rapid frequency hopping where a ramp function is approximated by a staircase step function the signal can be written as:

$$s_h(t) = s_r(nT) \tag{2.5.8}$$

In other words, the rapid frequency hopping is giving a discrete time approximation to the frequency chirp.

$$s_h(nT) = m(nT)e^{j2\pi(f_o nT + k(nT)^2/2)} \tag{2.5.9}$$

This formula is valid for when the frequency acceleration constant is small enough such that dwell time is not larger than T.

# CHAPTER THREE

## 3.1 System Design

The proposed system employs an agile synthesizer that can be used in the transmit and receive modes in a half duplex frequency hopping radio. Fast hopping requires a fast frequency synthesizer in the transmit mode and an efficient method for synchronization in receive mode.

The agile synthesizer relies on one synthesizer being locked while the other is being re-tuned. In transmit mode one LO is used for transmitting data on the current channel while the other LO is being tuned to the next hopped channel. Once the synthesizer is stable for a time $T_d$ the second LO is switched in. Across the hop packet, integrity can be maintained by stuffing several bytes into the data so that spurious noise does not destroy packet data across the hop. Fig. 3.1.1. shows a transmitter configuration using the agile synthesizer and a quadrature modulator.

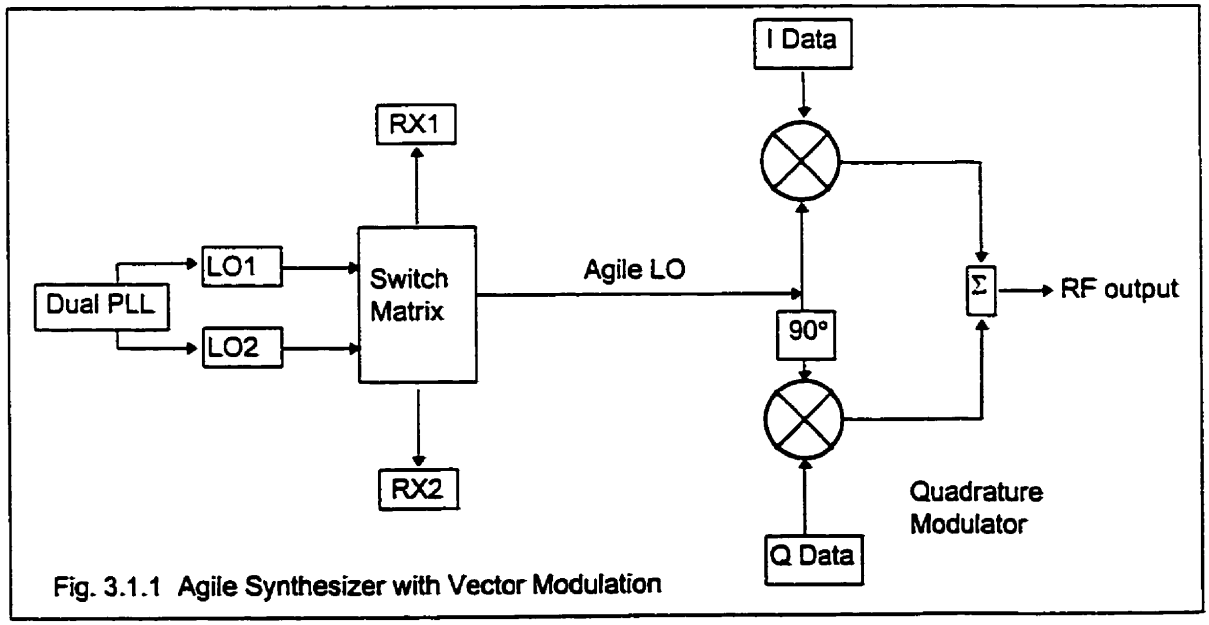



Fig. 3.1.1 Agile Synthesizer with Vector Modulation

A quadrature modulator permits the implementation of different modulation schemes including GMSK, QPSK, and QAM. A simple FM scheme can be implemented by applying the modulating signal to the VCO tuning port, however, this method requires the use of a manchester coding scheme. The modulating signal needs to be applied to both tuning ports of each LO in the agile synthesizer.

The receive mode relies on passive synchronization to remain locked to the transmitter. Fig. 3.1.1 shows the two LOs with one connected to RX1 and the other to RX2. The dwell time is chosen so that both LOs are locked around the hopping window. This allows the receiver to listen to the current channel and to the next hopped channel simultaneously. By using the LOs in a leap-frog manner synchronization is easily maintained across a hop.

By monitoring the RSSI and a correlation pattern on both channels the transition from one channel to the next is completed with a small dwell time and little preamble overhead.

Fig. 3.1.2 shows a typical structure for the receiver implementation using the dual LO supplied by the agile synthesizer. This receiver has a common front end but two separate IF stages. There is an isolation amplifier and an LO rejection filter which reduces the LO feedback coupling between the two IF sections. The LO leakage between the two IF stages must be low enough to ensure that a spurious free dynamic range is maintained with good adjacent channel rejection and image rejection characteristics.

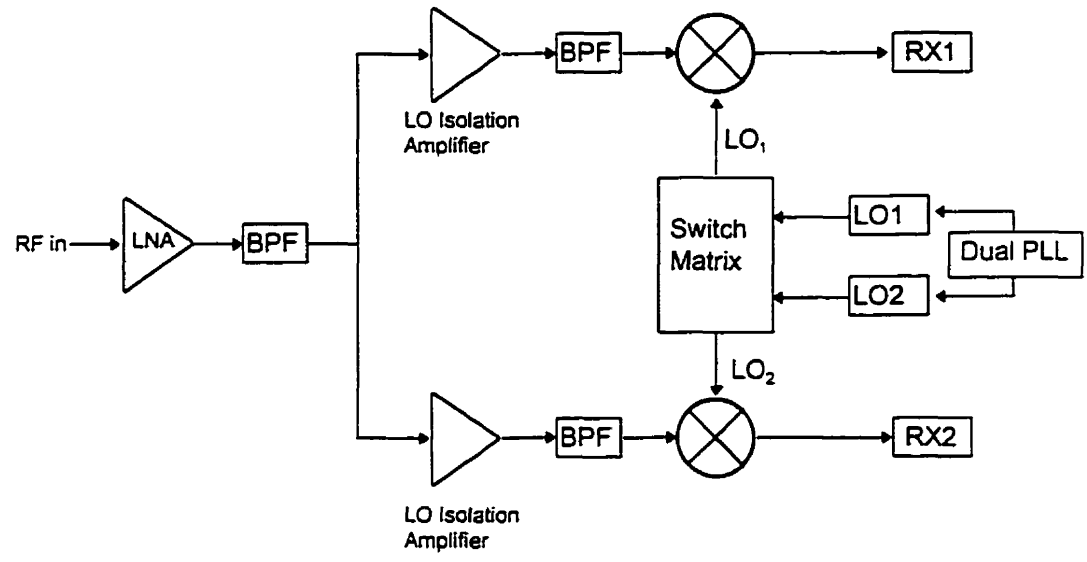



Fig. 3.1.2 A Typical Receiver Implementation Using the Agile Synthesizer

The IF stages will implement the demodulation to give received data and RSSI outputs. Since the receive channels are selected by $LO_1$ and $LO_2$, both IF stages can be implemented identically. The design of the IF stages is less complicated if an FM scheme is chosen over a quadrature demodulation scheme.

The analysis of system performance including noise figure, dynamic range and BER performance are beyond the scope of this investigation. Instead, the focus will be on the agile synthesizer since it is the key component to the design of this system.

## 3.2 Simulation of the Agile Synthesizer

The two local oscillators in the agile synthesizer do not maintain phase coherence across the frequency hop. In this section, a Matlab simulation compares the spectral output of a phase coherent and phase non-coherent agile synthesizers. The fast hopping is simulated by piecing together two vectors of different frequencies. Phase coherence can be maintained by adjoining the two vectors at the zero crossing. In the simulation the two channels are 1MHz apart and the sampling frequency is 3 MHz.

The switch matrix will cause spurious emissions when switched between $f_1$ and $f_2$. Fig 3.2.1 shows the spectral output with no carrier modulation. RF splatter and spikes appearing are separated from the carrier by the switching frequency. Non-coherent phase between the two oscillators is simulated by using two independent random phase variables with a flat distribution from $0 \leq \phi < 2\pi$.

Fig 3.2.2 shows the simulation results where phase coherence is maintained across the frequency hop. Clearly, spurious emissions are reduced by 15 to 20dB with phase coherence.

The switch matrix has the same effect as does a mixer. Practically, hops will be every 1ms, which creates spurious emissions at 1kHz increments from the carrier. Faster dwell time will be possible when a Fractional-N dual synthesizer is used.

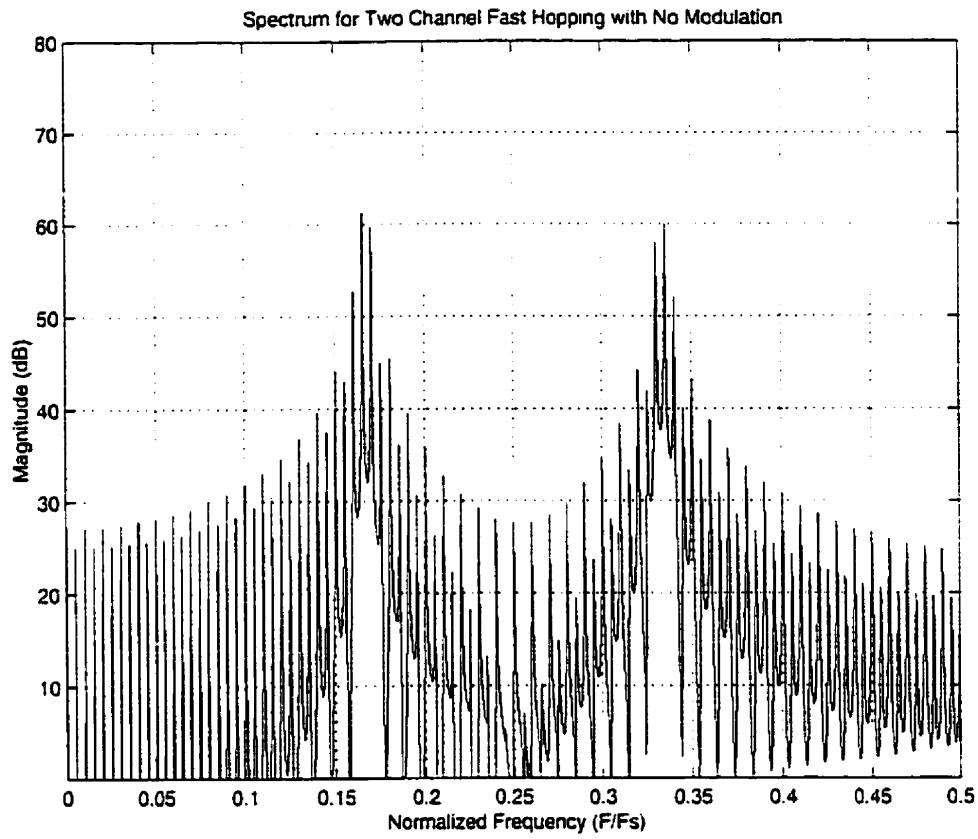



Fig. 3.2.1 Two Channel Fast Hop with Discontinuous Phase

The second part of the simulation FM modulates the agile LO with 100kHz deviation. No pulse shaping is implemented in the simulation. Practically, pulse shaping such as raised cosine or guassian can be implemented using a data filter prior to modulation. Furthermore, a post modulation filter would reduce adjacent channel power leakage.

The modulated carrier is passed through the channel and FM demodulated in the receiver. The receiver uses the FM demodulation with a Butterworth filter implemented at a normalized frequency of 1/3.

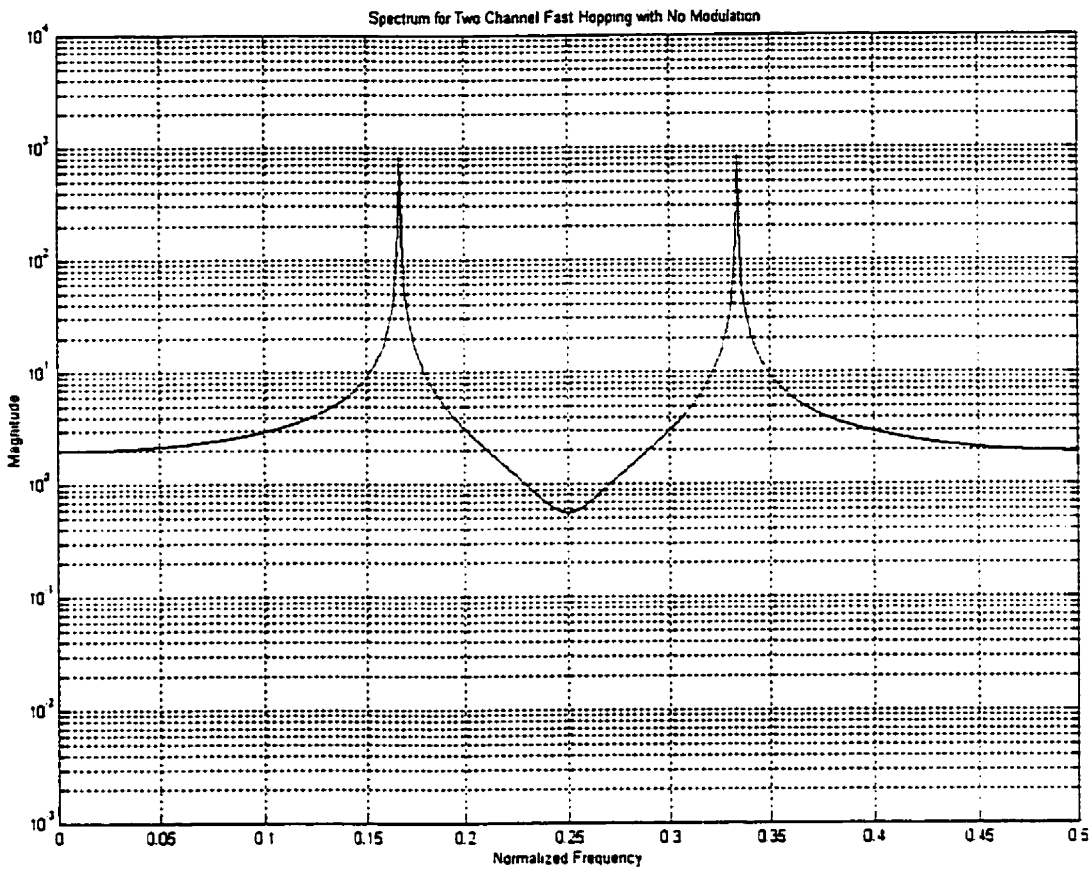



Figure 3.2.2 Two Channel Fast Hop with Continuos Phase

Other modulation and demodulation techniques can be used in this simulation. However, since phase coherence cannot be maintained across the hopping window the receiver will need to synchronize the phase of the demodulated signal after each hop. FM and differential phase modulation schemes are more suitable than non-differential phase modulating schemes.

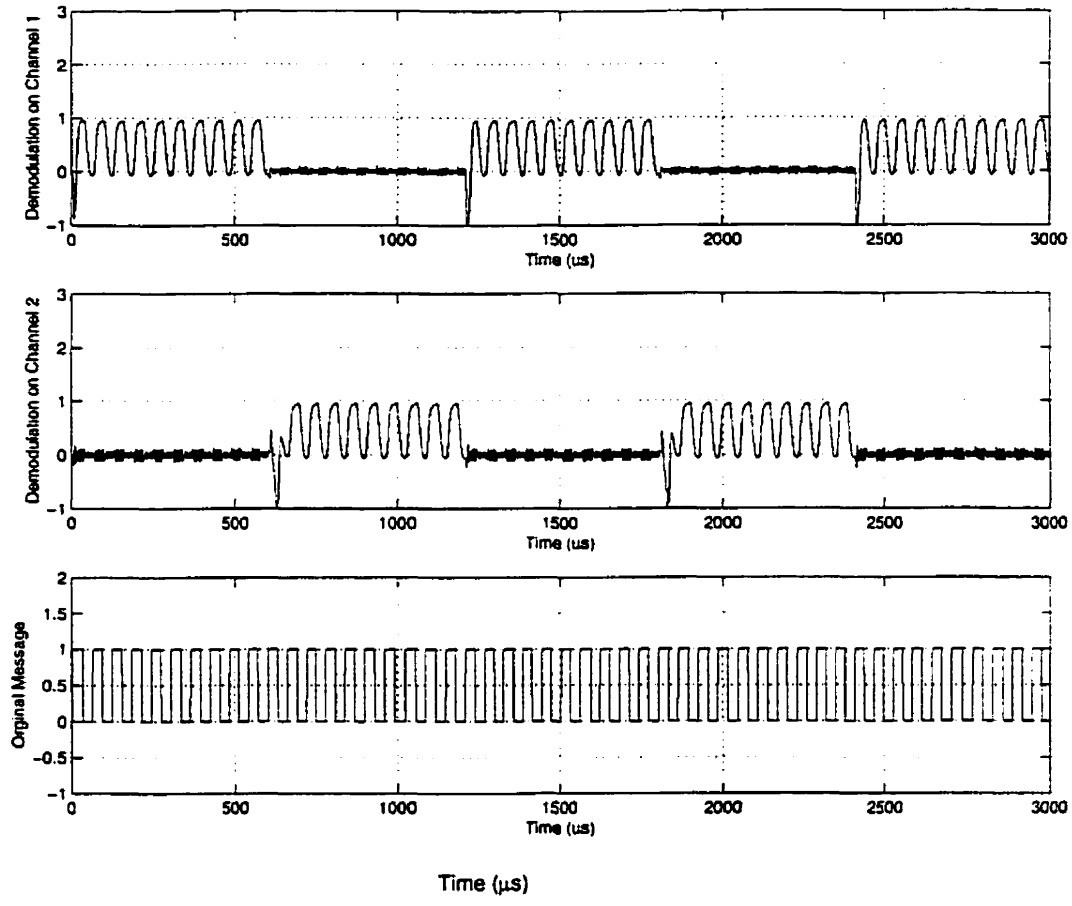



Figure 3.2.3 Demodulated Data for a Two Channel Agile Synthesizer

Figure 3.2.3 shows the demodulated data for a two channel agile synthesizer. The LO is switched between $f_1$ and $f_2$ which are 1MHz apart. There is no phase coherence maintained between the two frequencies. Data is fed into the frequency synthesizer continuously while the LO is switched by the agile synthesizer. Two receivers are implemented at the two channels. Figure 3.2.4 shows a close in view of the data. The bits near the hopping window suffer from some transient ripple; however, the adjacent received bits are recoverable. By adding buffer bits around the hopping window it may be possible to maintain packet synchronization across the frequency hop.

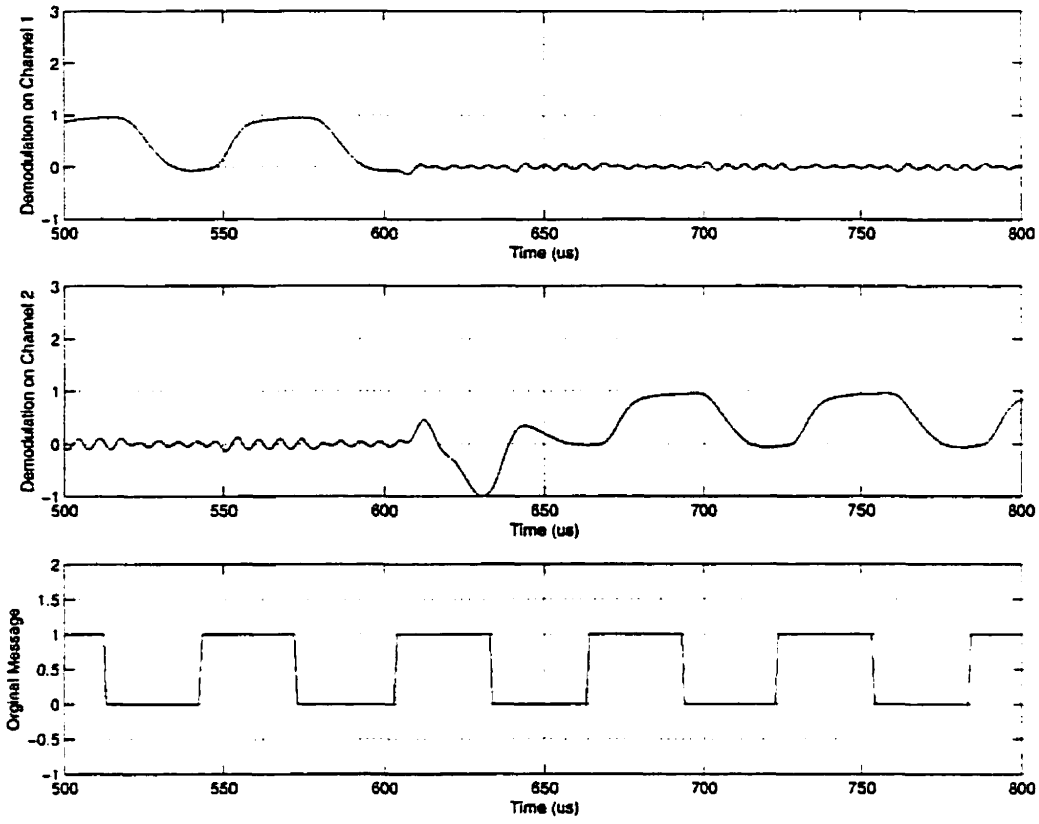



Figure 3.2.4 Demodulated Data Around the Hopping Window

This section shows that the concept of an agile synthesizer may allow the transmission of continuous data allowing synchronization to be maintained across a frequency hop for FM and differential phase shift modulation schemes. Although many implementation issues have not been addressed in this section, the agile synthesizer will work at least on a theoretical basis.

More simulation may be undertaken to study BER performance in different multipath channels. This is beyond the scope of this investigation and with some theoretical simulation in hand, we shift focus onto the hardware implementation of the agile synthesizer.

## 3.3 Design of the Agile Synthesizer

The switch matrix would be used in transmit and in receive modes. In transmit mode the switch matrix selects one of the two LOs, and in receive mode it channels the corresponding local oscillator to its receiver.

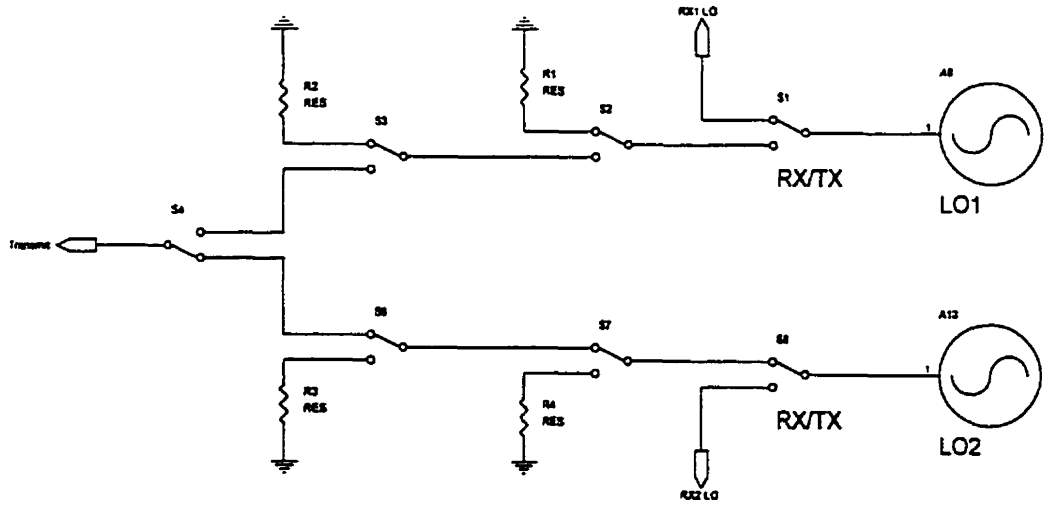



Figure 3.3.1 Switch Matrix Block Diagram

Figure 3.3.1 shows the block diagram for the switch matrix. Several stages are used to obtain sufficient isolation while increasing the insertion loss. During RX mode the SW1 switches LO1 to the RX1 and SW2 switches LO2 to RX2.

The four control lines that drive the switch matrix are TX_mode, RX_mode, LO1_Select, and LO2_Select. Selecting RX_mode and negating TX_mode actives SW1 and SW8 to supply the LO to the RX sections. In this configuration, the LO1_Select and LO2_Select do not matter.

For transmit mode TX_mode is asserted and RX_mode is negated, while LO1 and LO2 select lines determine which LO is channeled to the output. The switches SW2 and SW3 function to increase the isolation between local oscillator coupling. Similarly, SW6 and SW7 increase isolation for LO2. The final switch in the transmitter chain is SW4. This switch connects both LO feeds to the transmit LO. Table 3.3.1 summarizes the modes of operation for the switch matrix.

Table 3.3.1 Modes of operation for the Switch Matrix.

| Mode | RX_mode | Tx_mode | LO1_Select | LO2_Select |
|---|---|---|---|---|
| Receive Mode | **Asserted** | **Negated** | **Negated** | **Negated** |
| Transmit LO1 | **Negated** | **Asserted** | **Asserted** | **Negated** |
| Transmit LO2 | **Negated** | **Asserted** | **Negated** | **Asserted** |

The resistors shown in Fig. 3.3.1 act as matched termination load on the outputs of the isolation switch. This reduces reflection and radiation of the local oscillators throughout the circuit.

### 3.3.1 Loop Filter Design

The agile synthesizer is based on the local oscillator design. The LO signal is generated using a dual synthesizer which can be analyzed as a single oscillator at a time.

The charge pump current and the tuning sensitive are given in the datasheets [3] [19]. The charge pump current is 5mA for the LMX-2320 and the tuning sensitivity is 43MHz/V. Rewriting these values in radian form gives:

$K_{\phi} = 5/(2*\pi)$ mA/rad
$K_{vco} = 43*(2*\pi*10^{6})$ rad/V
N = 9350

Using a reference frequency of 100kHz and a phase margin of 45°, equations 2.2.1.5 to 2.2.1.12 can be used to give the following parameters.

R2 = 910 Ω
R3 = 18 kΩ
C1 = 15 nF
C2 = 0.1 uF
C3 = 470 pF

The loop filter configuration is shown in Fig. 3.3.2.

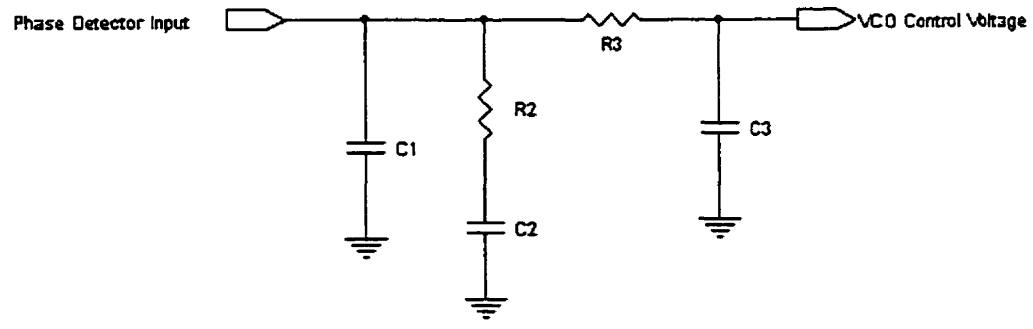



Fig. 3.3.2 Loop Filter Configuration

The loop filter has the following transfer function:

$$G(s) = \frac{9.1\text{x}10^{-5}s + 1}{1.15479\text{x}10^{-17}s^3 + 2.3379\text{x}10^{-12}s^2 + 1.15\text{x}10^{-7}s}$$

The general form of the open loop transfer function for the open loop response of the phase lock loop is:

$$\frac{\Theta_i}{\Theta_e} = \frac{K_\phi K_{vco} G(s)}{sN}$$

Substituting values into the general form gives:

$$\frac{\Theta_i}{\Theta_e} = \frac{19.565s + 215000}{s^2(1.07972865\text{x}10^{-13}s^2 + 2.1859365\text{x}10^{-8}s + 1.07525\text{x}10^{-3})}$$

This represents the open loop response for the PLL used in the synthesizer. The bode plot is shown for the PLL loop. The phase margin is 38.2° and the gain margin is 18.7dB. This means the closed loop function is stable. Furthermore, the suppression of the reference frequency at 100kHz is 60dB.

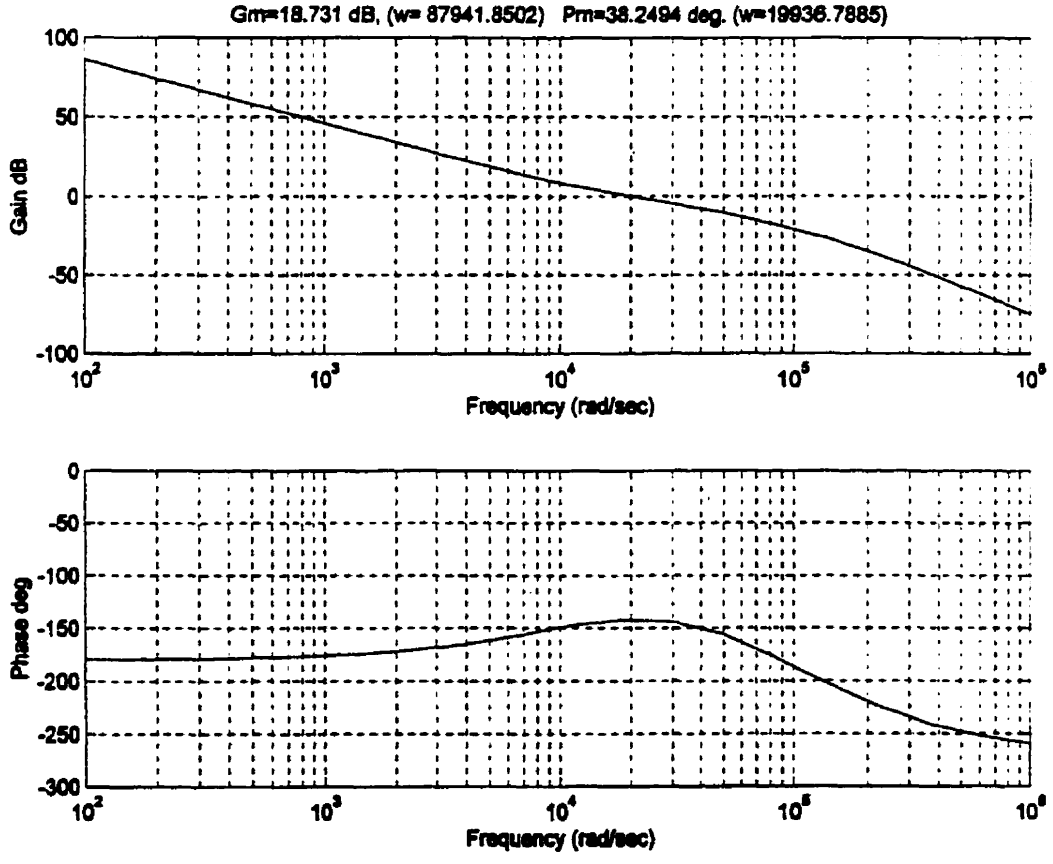



Fig. 3.3.3 Bode Plots for Phase Lock Loop

### 3.3.2 Agile Synthesizer Schematic

The schematic is drawn using Orcad Capture and the board layout is designed using Orcad Layout. The schematic pages are shown in Fig. 3.3.3 to Fig. 3.3.5. In addition to the agile synthesizer the schematics for the digital control section are shown.

The UPC132G is an individual GaAs switch which is used to build the switch matrix. The switch matrix was tested for isolation and insertion loss, and the manufacture specifications were confirmed (see Appendix A for specifications).

Each switch is connected according to the block diagram shown in Fig. 3.3.1. The switch sections are interconnected through a 100pF capacitor so that input and output ports of each section can be independently biased. Termination resistors are 50 Ω which act as an excellent match to the output ports. In addition, termination ports are also AC coupled with a 100pF capacitor. Each switch is controlled with two digital lines. The control lines are lowpass filtered using a resistor capacitor network (1KΩ and 100pF) to reduce cross-coupling between each switch section.

The digital section uses a PIC16C74 microcontroller to program the synthesizer and to provide a user interface through a host terminal. Level shifting circuitry is provided by U11 which is an ADM232L. This device converts TTL levels into RS-232 levels and vise-versa. The microcontroller is clocked 20MHz. The instruction execution time is generally 4 clock cycles or about 200ns. This is fast enough for the requirements of this investigation.

The frequency synthesizer is designed using the LMX2335TM which is a dual PLL synthesizer IC. The loop filter is already designed in section 3.3.1 and is used for both LOs. The synthesizer power and charge pump supply voltage lines are RC filtered to reduce spurious emissions and cross-coupling between the two LOs. A 27Ω resistor, 4.7uF capacitor, and 1nF capacitor provide the power supply decoupling from the 5V supply rail. The synthesizer is referenced from a 16MHz clipped-sine TCXO. The TCXO provides good temperature stability (2.5 ppm) and tight frequency tolerance. The synthesizer is programmed using a 3-wire interface connected to the microcontroller's SPI port. The lines are serial clock line, data line, and a chip select line. These three high speed lines are filtered to reduce digital ringing which may cause programming errors. The exact programming format is detailed in LMX2335TM datasheet [20]. An active RC filter is used to generate a digital lock detect circuitry. The lock detect indicates when the synthesizer has settled to the final frequency.

The output of the agile synthesizer is filtered with a 902 - 928 MHz bandpass filter. This ceramic filter provides good rejection of the harmonics generated by the LO. The synthesizer uses a 900 - 975 MHz Zcomm VCO (V580C06). The VCO U13 and U14 are four port devices. Two connections provide 5V and ground ports and the other ports are used for Vtune and RFout.

From each VCO a feedback path is provided to close the phase lock loop. The feedback lines are impedance matched to the synthesizer input port fin.

For this investigation an unregulated 12VDC wall adapter is used. This supply is linearly regulated on the board down to 5VDC. This will provide power for all the sections on the board.

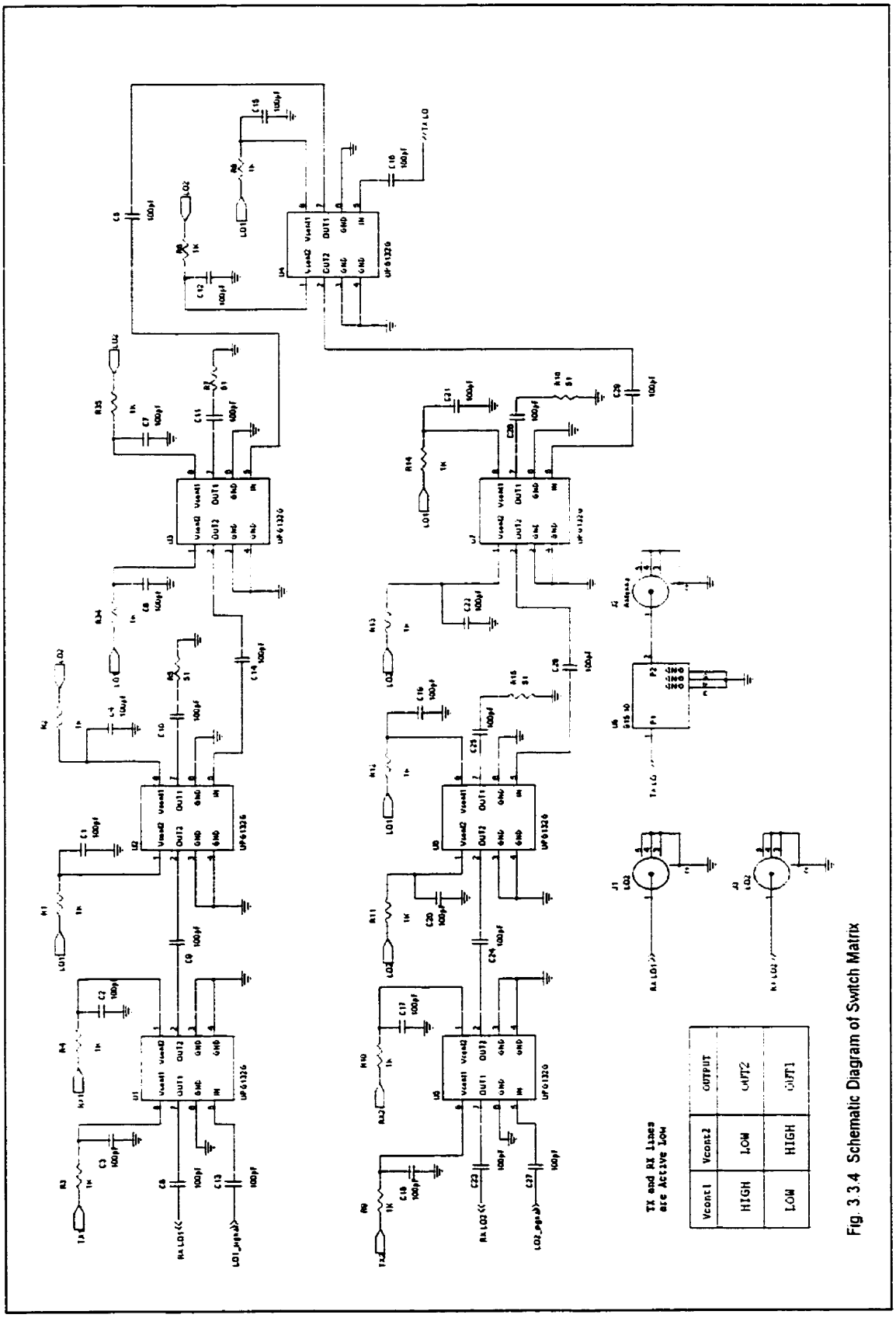



Fig. 3.3.4 Schematic Diagram of Switch Matrix

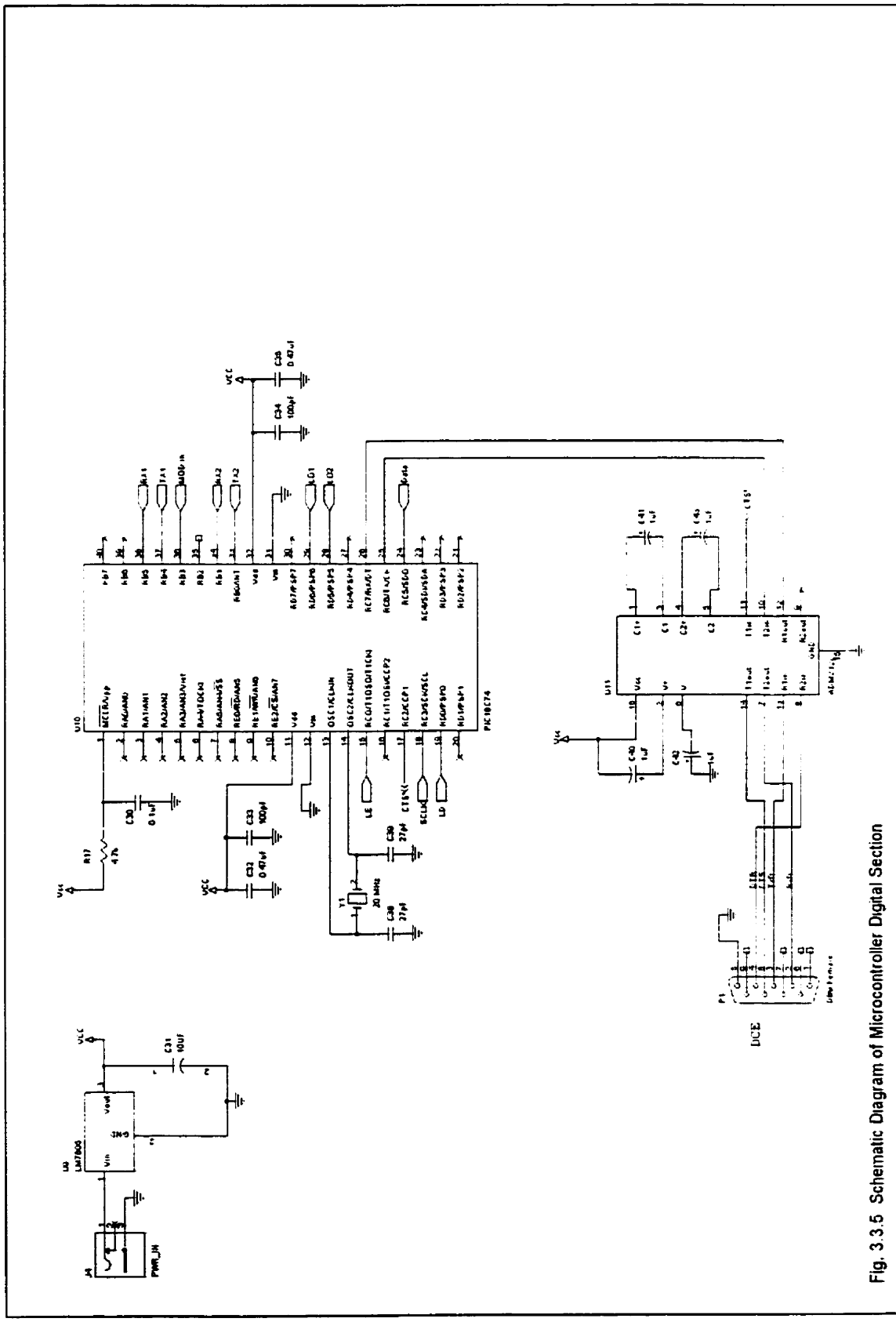



Fig. 3.3.5 Schematic Diagram of Microcontroller Digital Section

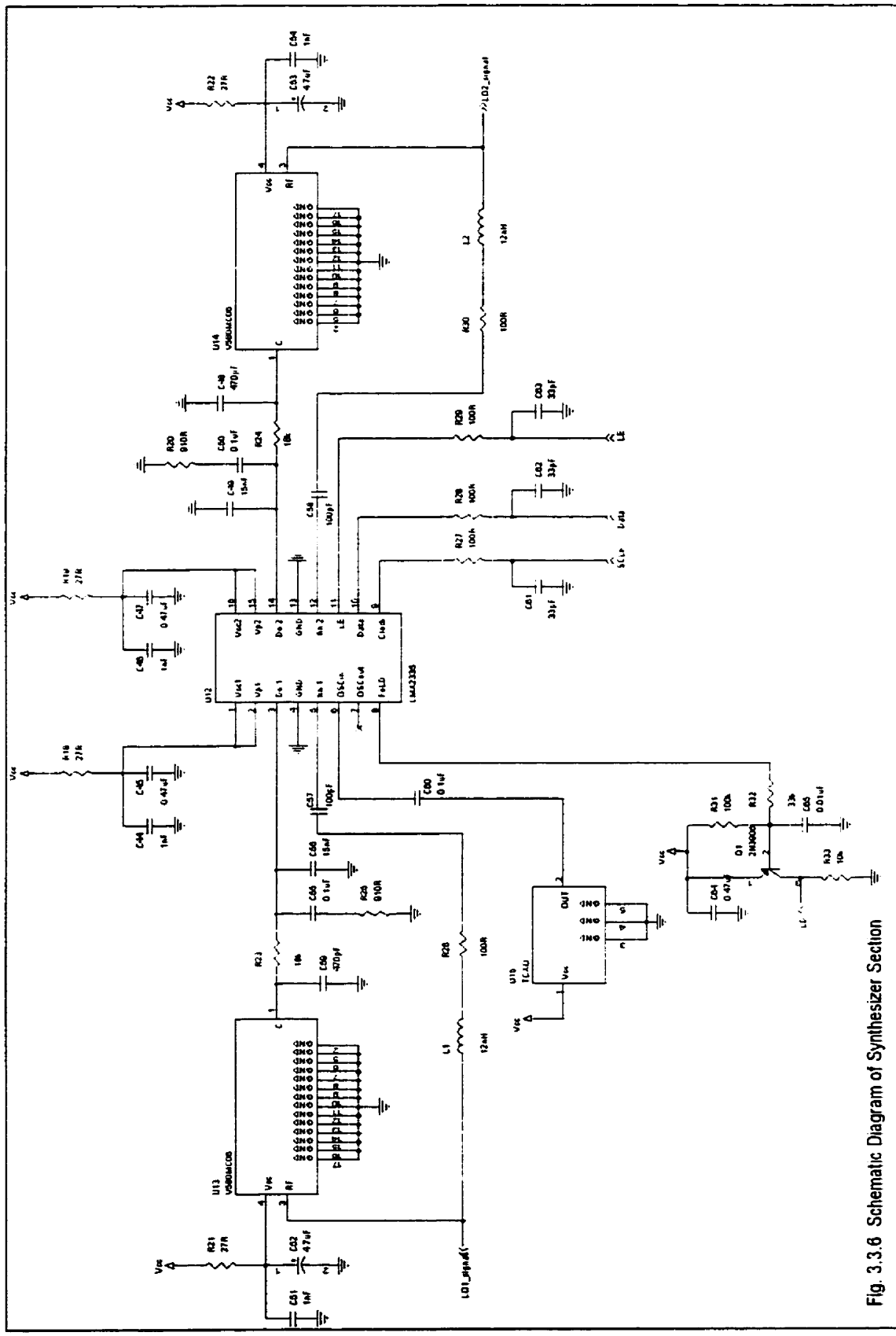


Fig. 3.3.6 Schematic Diagram of Synthesizer Section

## 3.4 Agile Synthesizer Board Layout

Figure 3.4.1 shows the top layer of the Gerber file for the agile synthesizer. Key sections are labeled.

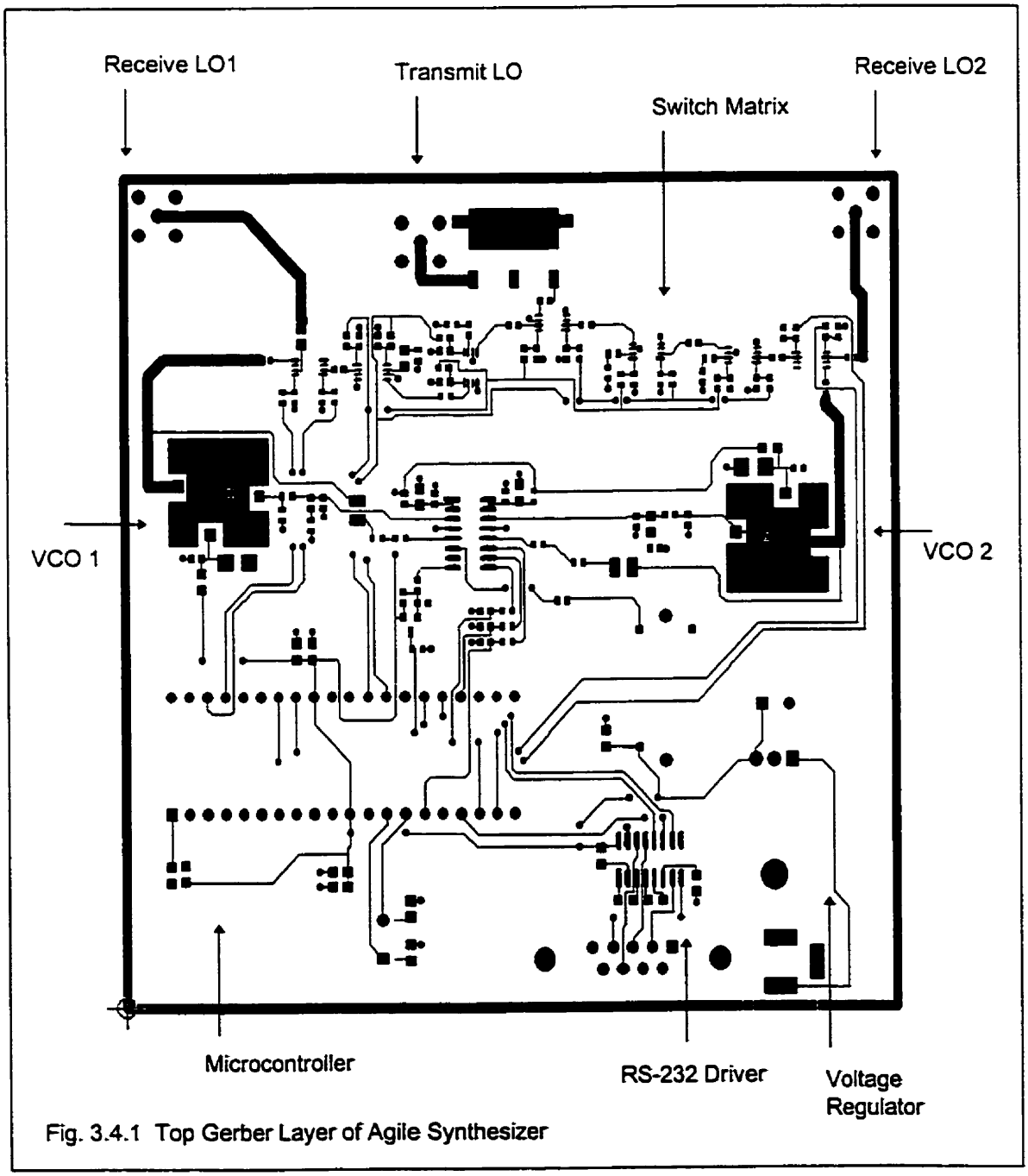



Fig. 3.4.1 Top Gerber Layer of Agile Synthesizer

The board used is FR4 with a dielectric constant of 4.5 and a 31 mils substrate height. The copper plating is 1.4mils or 1oz copper. Thus a 50Ω microstrip is approximately 58 mils wide. Since these boards undergo a solder reflow process the final microstrip will have a higher loss than would be expected with solder masked boards.

Trace widths are designed so that an impedance match is maintained between components. For short traces, thinner traces are more practical and introduce insignificant impedance mismatches. The gerber plot shows control lines from the microcontroller to the switch matrix. Although these lines are extensively filtered the layout placement is critical since RF coupling caused by control lines and power rails will significantly degrade the performance of the agile synthesizer.

The digital section is separated from the analog section in the bottom and top half of the board, respectively. Decoupling capacitors are placed physically close to the pins of the components which they decouple, thus reducing the inductive paths for high frequency signals. On the power rails, multiple decoupling capacitors are used so that a good path to ground is provided for a wide frequency range.

In accordance with good RF layout techniques the ground plane is not cut near microstrip lines. This reduces reflection and maintains a low VSWR across long microstrip lines.

The bottom gerber layer is mainly left as a solid ground plane (see Fig. 3.4.2). This allows the design of microstrip lines in the RF section on the top layer. Furthermore, a solid ground plane reduces spurious emissions by providing a better ground return path.

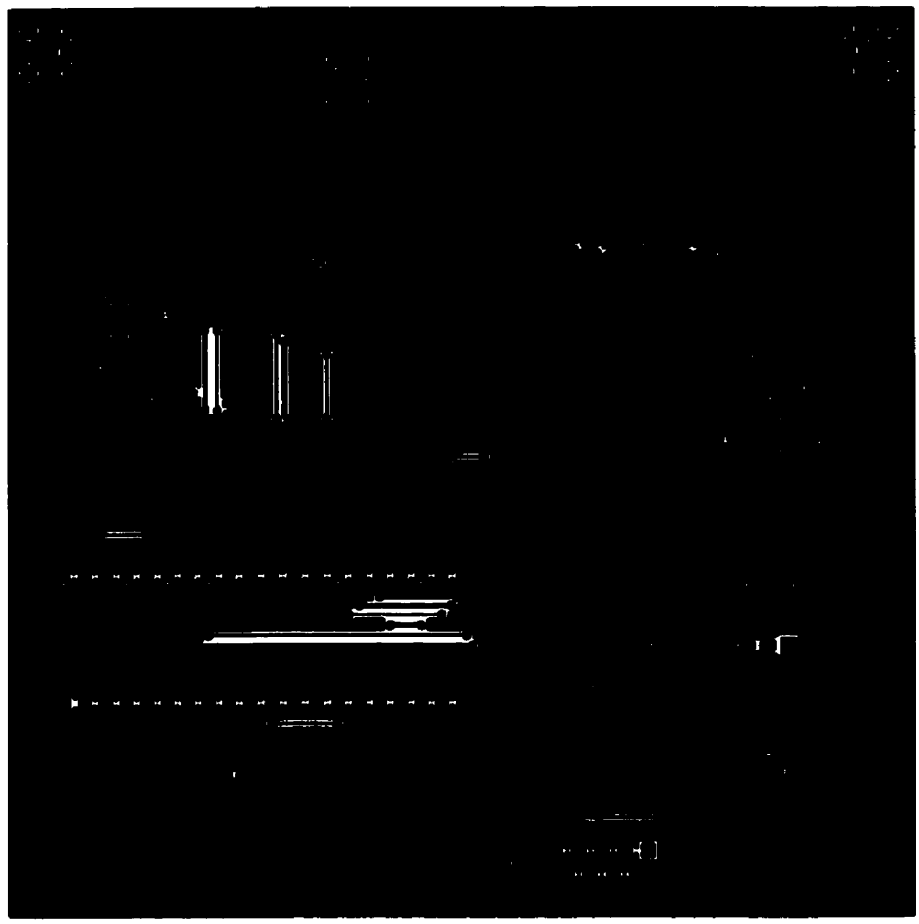


Fig. 3.4.2 Bottom Gerber Layer of Agile Synthesizer

The digital and analog section are not isolated grounds. Isolated grounds would improve digitally coupled noise in the agile synthesizer.

Figure 3.4.3 shows the component reference designators. This can be used with the schematic diagram to locate all components. Only the top and bottom metal layers were submitted to a board house for fabrication on a double sided board.

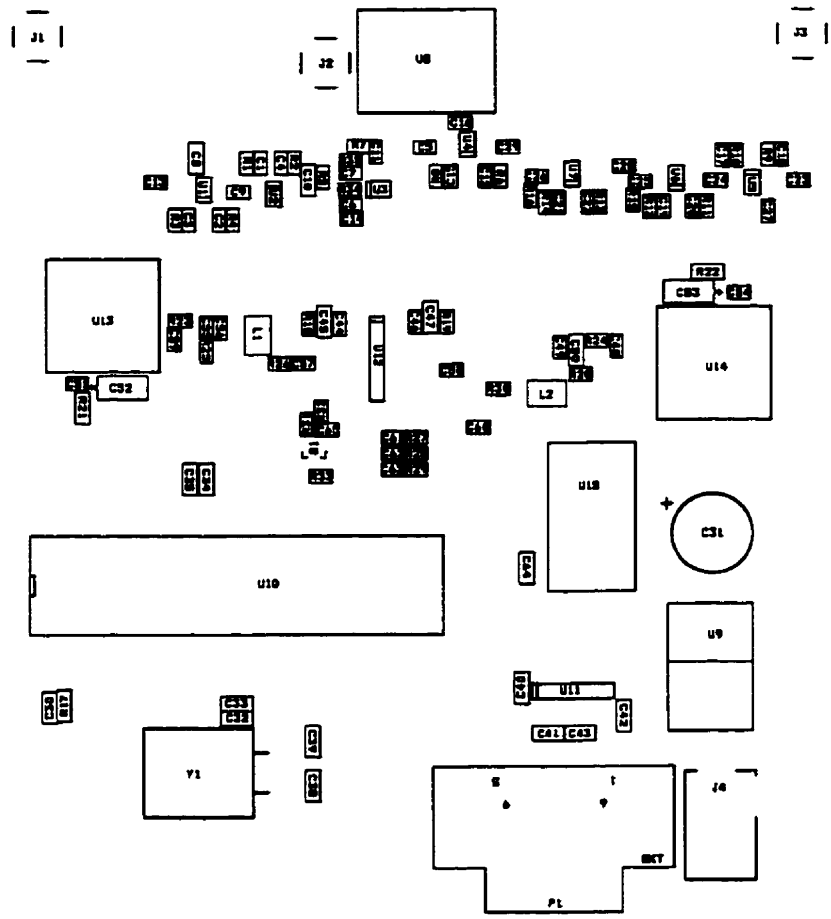


Fig. 3.4.3 Silk Screen Gerber Layer of Agile Synthesizer

## 3.5 Test Setup

Before the performance of the hardware can be tested code needs to be written for the microcontroller. A simple program is written to initially program the synthesizer to a single frequency. For source code listing refer to APPENDIX B. The code written for the PIC16C74 is compiled using the PICC compiler developed by Custom Computer Software Inc. Throughout the testing phase the code is modified to test hardware functionality including circuit board trace integrity.

Figure 3.5.1 shows a simple test bench sketch. A computer using a terminal program with a serial interface at 9600 baud is required to communicate with the microcontroller.

Calibration begins with the cables. The RF cables are initially calibrated for loss at 915MHz (center of the ISM band) using a reference source signal generator. There is no phase calibration required since the experiment will only look at the magnitude response.

Both LOs in the agile synthesizer were tested for its spurious output on the first, middle and last channel in the hopping range. Verification plots were performed on the selected frequency.

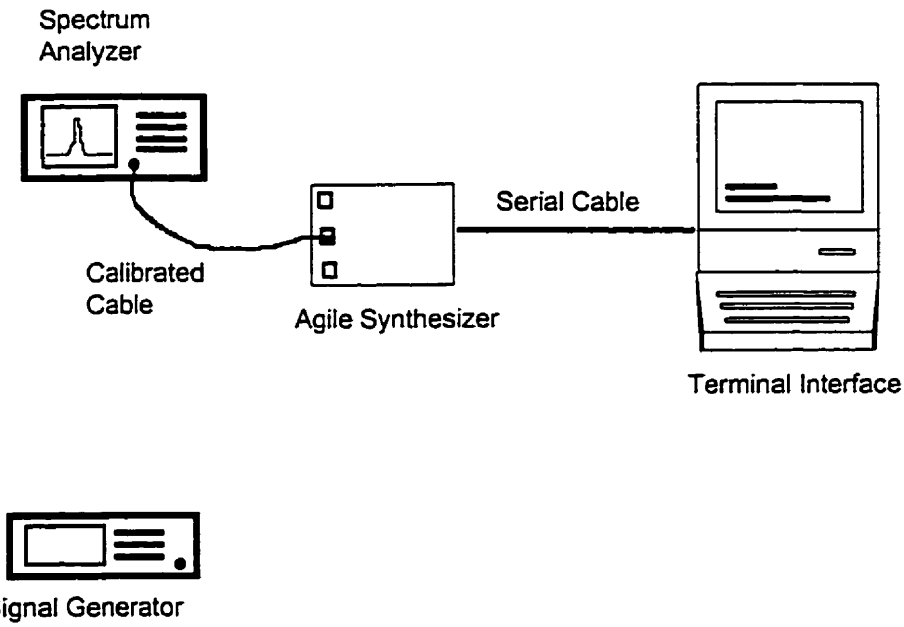



Fig. 3.5.1 Test Bench Layout

The following chapter examines some of the results of this investigation including the performance of the agile synthesizer in steady state and transient response.

# CHAPTER FOUR

## 4.1 Measurement Frequency Synthesizers

The dual synthesizer was tested on both sides. The oscillator frequency was 16.000MHz and the programmed channel was 902.3 MHz. The N divider was programmed to 160 giving a loop reference frequency of 100kHz. For exact details on programming the synthesizers can be found in National Semiconductor datasheet for the LMX2335TM [20].

The calibration of the cables showed that the cables had an insertion loss of 1.10dB and the DC block capacitor had a loss of 0.35dB at 915 MHz. This gives a total loss of 1.45 dB. All absolute magnitude measurements need to be adjusted by 1.45 dB.

The plot in Fig. 4.1.1 was taken from an HP8594E spectrum analyzer. This plot is taken at the output of the VCO. The exact frequency of the LO of 902.3 MHz was not generated. Upon further investigation it was found that the TCXO reference frequency was 16.000335MHz. The offset of 335 Hz causes a small error in the loop reference frequency. Loop reference frequency is calculated as:

$$F_{ref} = \frac{16.000335}{160} = 0.10000209375 MHz$$

with N = 9023 for this particular channel the final synthesized frequency is $NF_{ref}$

$$N \cdot F_{ref} = 902.31889 MHz$$

This frequency matches exactly the value seen on the spectral output. However, since the TCXO frequency reference is the same for all the tests any offset biases are eliminated.

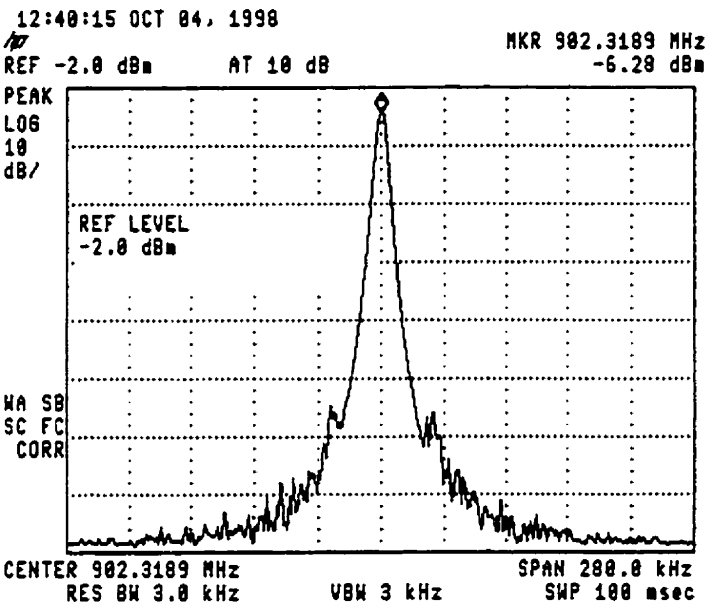



Fig. 4.1.1 Synthesizer Output Spectrum 280KHz Span

Looking at Fig. 4.1.1 the reference sidebands are sufficiently rejected with more than 60dBc. The synthesizer also shows good suppression of spurious emissions. This indicates that the layout and the grounding around the synthesizer is acceptable. This shows that the phase lock loop is stable and has sufficient sideband suppression. The calculated and measured rejection are similar. The third order loop filter results in longer lock times than a second order loop filter.

Figure 4.1.1 shows that the output power is -6.26dBm which is corrected to -4.81dBm with the cabling loss. This is ample output power and may be resistively tapped to decrease load pulling on the VCO.

A closer look at the local oscillator shows good phase noise response. The phase noise can be calculated from the spectral output using the equation below:

$$N_{phase} = C + 10\log(RBW)$$

From Fig. 4.1.2 the phase noise is -64.7 dB/Hz at 2.7kHz and from Fig. 4.1.1 the phase noise is about -110dB/Hz at 100kHz offset. This is close to the performance of the VCO alone [19].

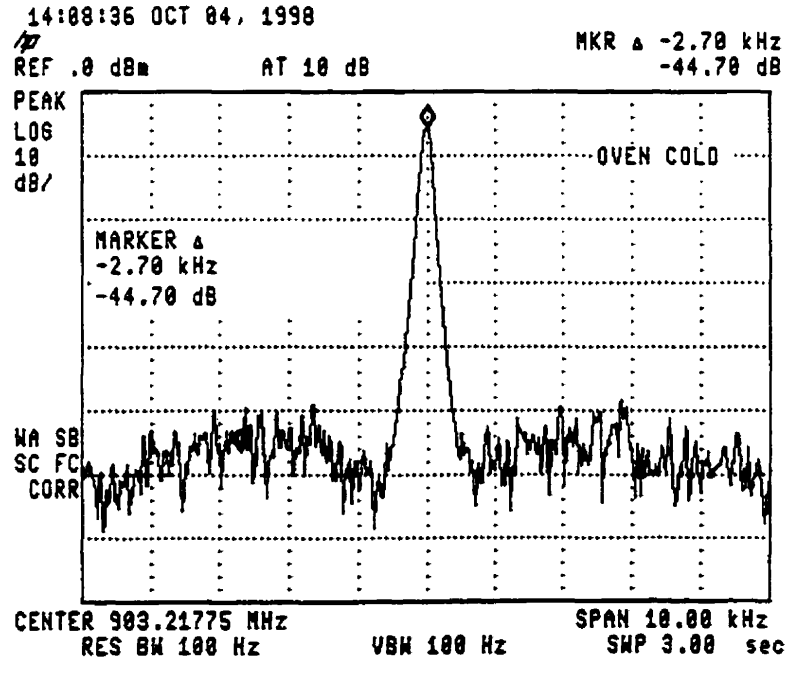



Fig. 4.1.2 Synthesizer Output Spectrum 10KHz Span

Thus the synthesizer designed shows good performance for phase noise, spurious response, sideband suppression, and output power.

## 4.2 Static Measurement of Switch Matrix

The switch matrix was tested by writing code to drive the control lines and program the synthesizer. The laboratory equipment used to test the prototypes were a HP ESG-3000A signal generator and HP 8594E spectrum analyzer.

In the first test the agile synthesizer is configured to transmit LO1 without switching to LO2. A spectral plot is shown in Fig. 4.2.1. The isolation between the two oscillators is measured at 57.7 dB. The test was repeated, but this time the agile synthesizer is configured to transmit LO2 without switching to LO1. The results are similar.

These results are acceptable and comparable to the performance expected from a good quadrature vector modulator. The single sideband modulator has typically 40 dB of carrier and sideband rejection. Hence the agile synthesizer gives 15dB more rejection of the adjacent LO than a quadrature vector modulator does for carrier and single sideband rejection.

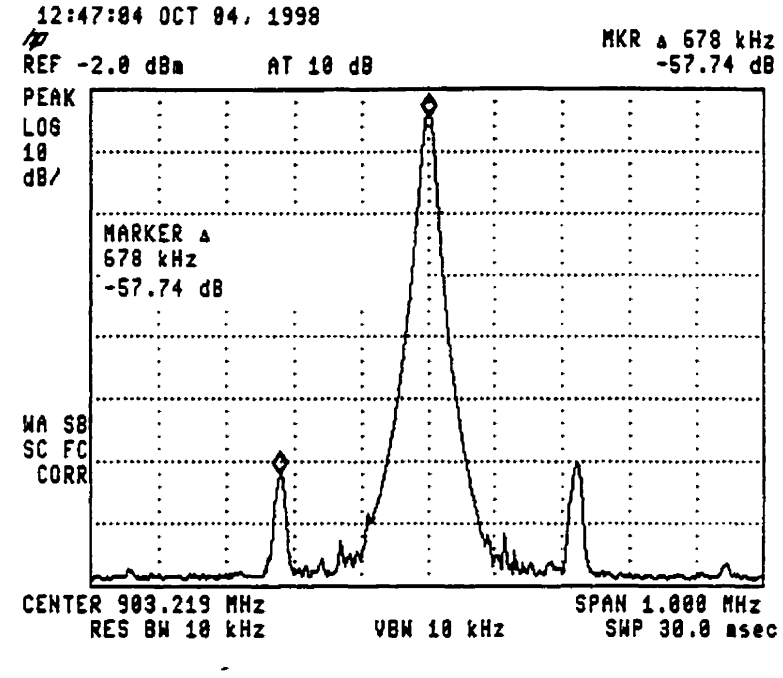



Fig. 4.2.1 Rejection of Adjacent LO of Agile Synthesizer

Although the results are practically acceptable, the measured isolation between LO1 and LO2 is actually lower than expected. RF coupling and leakage between board section accounts for the difference. The expected theoretical isolation is 80dB, however, tests show that the switch matrix has only 58 dB of isolation.

It is difficult to model leakage or cross coupling into a non-circuit simulation program such as Matlab. Thus this finite adjacent channel LO was not accounted for in the Matlab simulations. However, the results of the simulation are still valid since they were used to compare coherent vs non-coherent agile synthesizers.

Further investigation shows that the power lines provide a coupling path between LO1 and LO2. This path by-passes the switch matrix and hence decreases rejection. Independent power feeds using low drop out regulators as shown in Fig. 4.2.2 will improve power supply coupling. Using separate low drop out regulations for each section is a common practice in modern communications equipment.

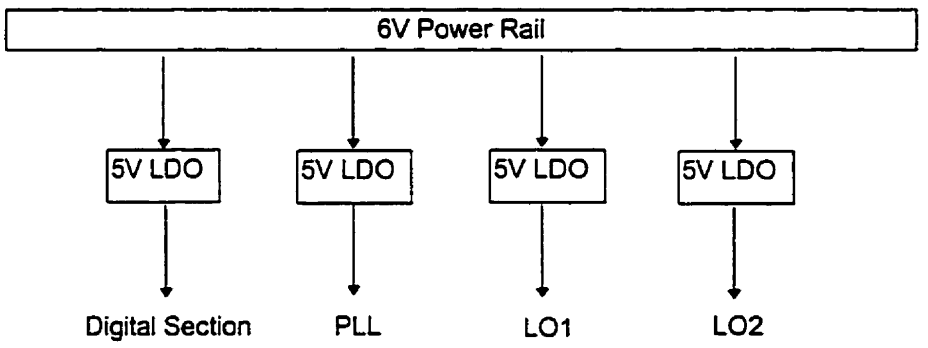



Fig. 4.2.2 Recommended Power Supply Isolation for Reduced RF Coupling

Control lines provide a coupling path for RF signals between LO1 and LO2. Although these control lines are lowpass filtered, layout changes will improve performance. By changing the board from two layers to four layers a solid ground plane can be implemented as shown in Fig 4.2.3. The RF signal layer would still be the top layer, layer 1. The power layer would be absent, layer 2. On the other side layer 3 would be the ground plane on which no control lines would run. The fourth layer is where the digital control lines would be placed. This would provide isolation between the RF routing layer and the control lines.

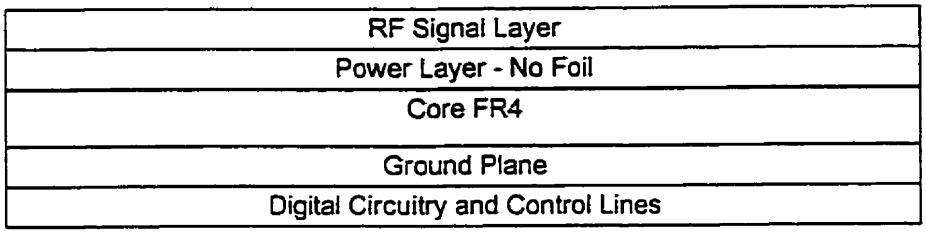



Fig. 4.2.3 Recommended Board Structure for Reduced RF Coupling

Additional improvements can be made to the actual RF microstrip traces by shortening the lines as much as possible. Microstrip lines which are solder reflowed have a higher tangent loss factor than solder masked traces. Each local oscillator may also be physically separated to increase isolation. This may be accomplished by using separate ground return paths and tin cans for shielding. In Fig 4.2.4 the top layer gerber is shown and indicators have been placed on the leaky lines coming from LO1 and LO2. Furthermore, note that these lines have sharp angle turns. These corners increase RF radiation from the microstrip. Using curved traces would give better performance.

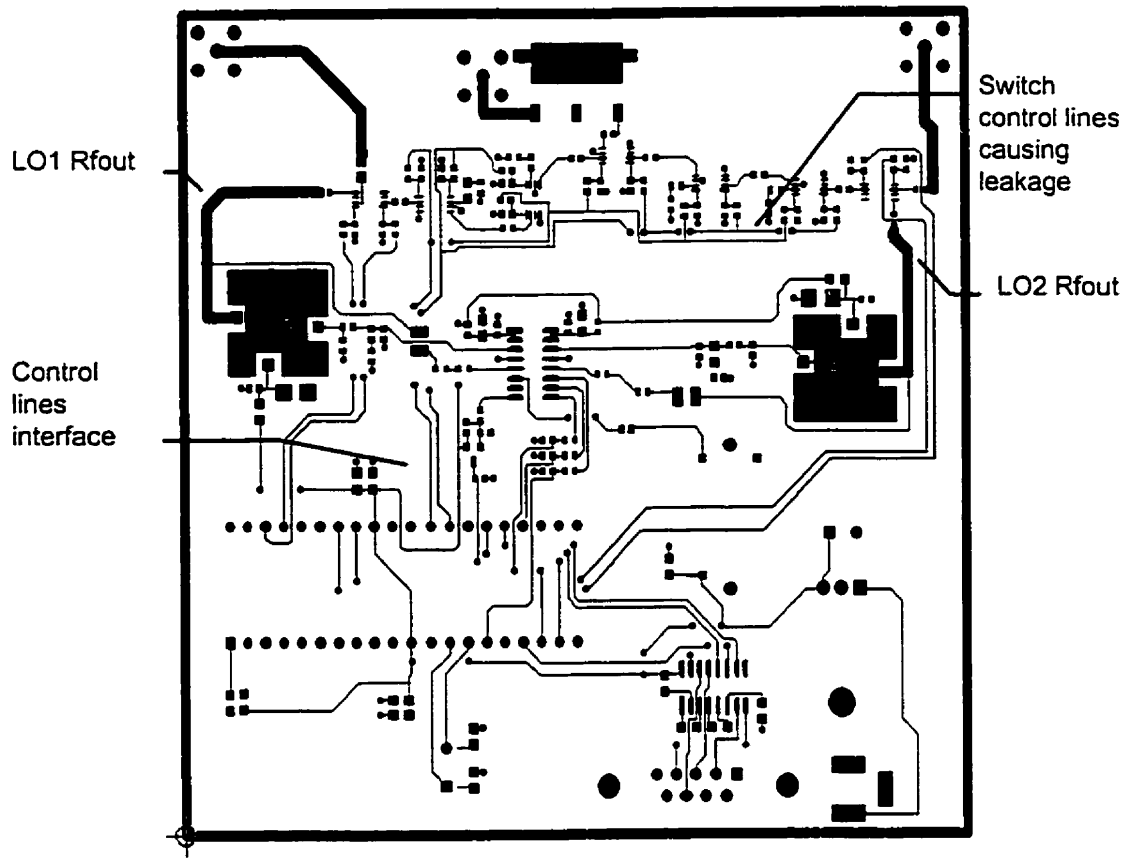



Fig. 4.2.4 Agile Synthesizer Board and Interface Circuitry

Direct coupling and leakage across the switch matrix cannot fully explain the spurious emission occurring on both sides of the carrier. Direct coupling should only give rise to a sideband on the same side as the adjacent LO is tuned. It is a non-linear transformation that causes both upper and lower sidebands.

Further investigations give evidence to support a second mode of coupling. A closer look at the upper and lower sidebands reveals that the sidebands actually are not exactly equal.

Although, there is not a large amplitude difference, repeated measurements show the spurious sidebands occurring at the adjacent local oscillator frequency are always higher. Fig. 4.2.5 shows that the lower spurious sideband is about 5dB higher than the upper sideband. The adjacent LO was tuned to the same frequency as the lower side band.

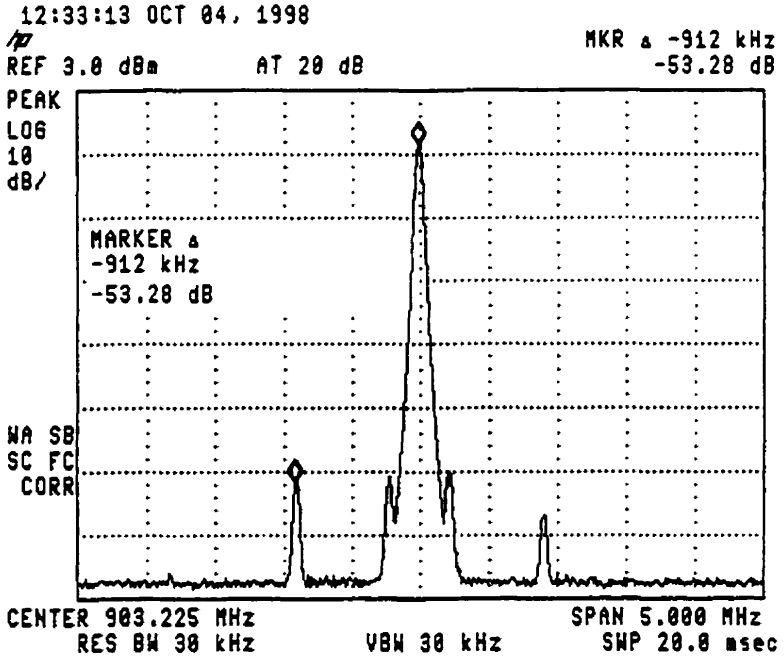



Fig. 4.2.5. Double Sideband of the Agile Synthesizer's Power Spectrum

In order to generate the upper sideband the system requires a non-linearity of third order or greater. The frequency difference between the two LOs is the beat frequency. If this is mixed in with the selected LO then the result is the upper and lower sidebands as seen in Fig. 4.2.5.

The transformation below shows the two sideband terms generated in a third order non-linear transfer function.

$$\begin{matrix}\cos(2\pi f_1 t + \theta_1) \\ \alpha\cos(2\pi f_2 t + \theta_2)\end{matrix} \xrightarrow{X^3} \begin{matrix}\beta_1\cos\{2\pi[f_1-(f_2-f_1)] + \varphi_1\} \\ \beta_2\cos\{2\pi[f_1+(f_2-f_1)] + \varphi_2\}\end{matrix}$$

However, a third order non-linearity may not have sufficient magnitude to generate the observed spurious frequencies. A more likely transformation would be a double second order transformation. In effect this is a fourth order non-linearity which was probably caused by the cascading of two second order transformations. Here are the terms of interest generated by this method:

$$\begin{matrix}\cos(2\pi f_1 t + \theta_1) \\ \alpha\cos(2\pi f_2 t + \theta_2)\end{matrix} \xrightarrow{X^4} \begin{matrix}\beta_1\cos\{2\pi[f_1-(f_2-f_1)] + \varphi_1\} \\ \beta_2\cos\{2\pi[f_1+(f_2-f_1)] + \varphi_2\}\end{matrix}$$

Non-linear performance can occur inside each VCO. The VCO output port is connected directly to an oscillating transistor running in saturation. RF signals present at the output port of the VCO can undergo some degree of mixing. This may account for a second order non-linear characteristic.

The UPC132G GaAs switch may also contribute to the non-linear performance. Although these tests are performed in static mode, the switches are not toggled, the FET inside the switch exhibit some non-linear characteristics at higher power.

## 4.3 Dynamic Measurement of Switch Matrix

The previous section discussed the static tests performed on the switch matrix. This section focuses on the dynamic performance of the switch matrix and of the dual synthesizer block. The first dynamic performance test is set-up by continuously tuning LO to the same frequency while switching between LO1 and LO2. The spectrum analyzer is set to the frequency of LO1 and the Span is set to 0 Hz. The sweep time is set to 50ms. Fig. 4.3.1 shows the carrier power as a time function where the agile synthesizer is hopped every 10ms.

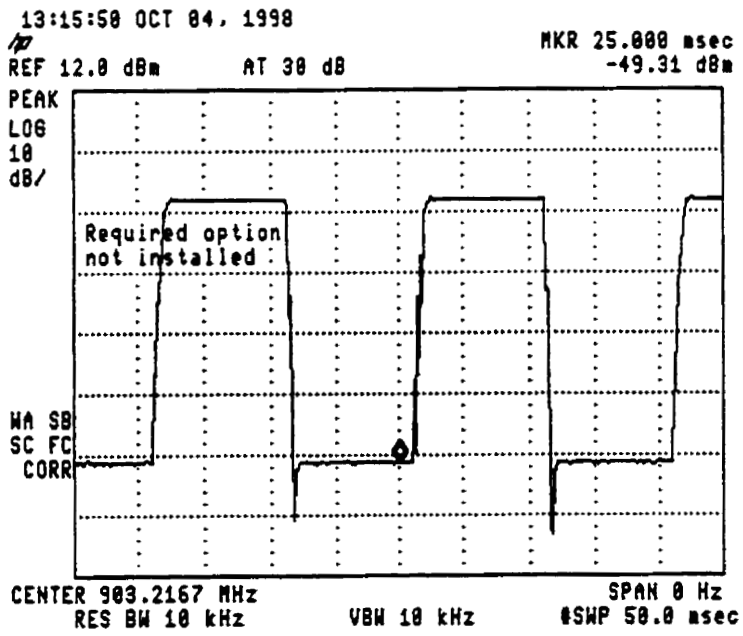



Fig. 4.3.1. Carrier Power on a Single Channel for 10ms Hops

A closer look at the carrier power rise time is shown in Fig. 4.3.2. The carrier does not show good rise time performance. There is a significant amount of RF ringing. This ring increases the rise and fall times and directly impacts the tuning time. This is serious since the entire purpose of the agile synthesizer is to provide a quick transition from one frequency to another - reduce tuning time. Furthermore, large ringing will cause out of channel emission to be excessively high.

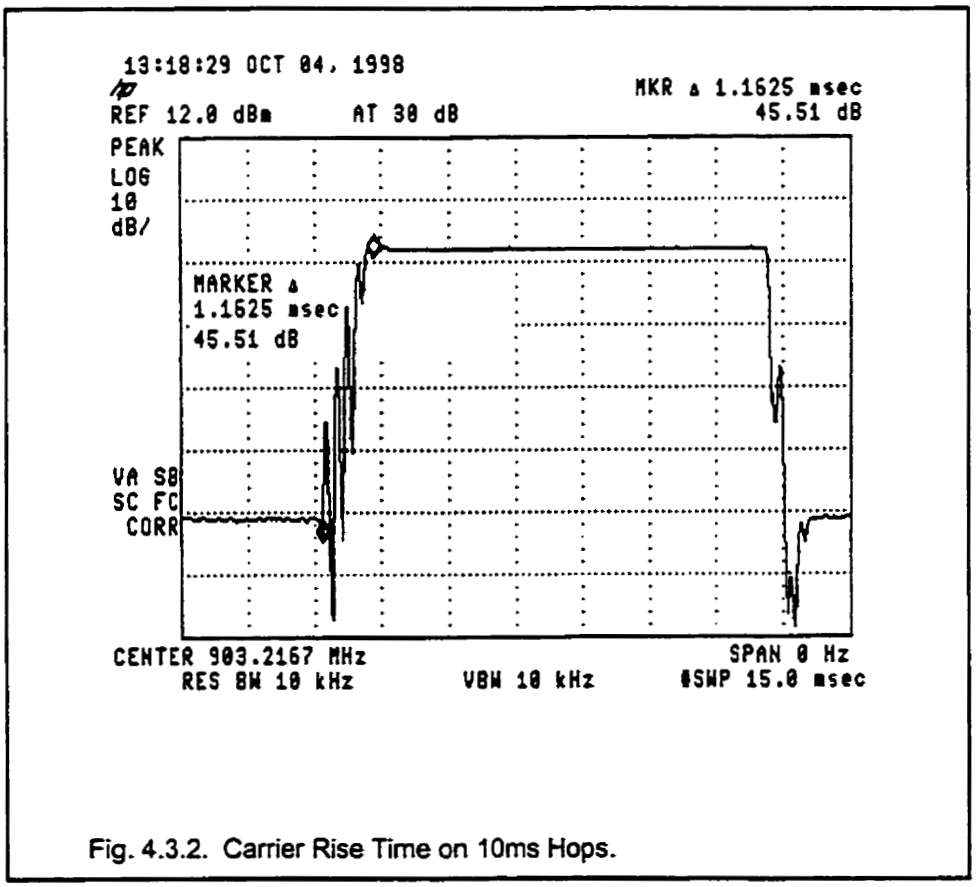



Fig. 4.3.2. Carrier Rise Time on 10ms Hops.

The rise time as seen in Fig. 4.3.2 does not agree with what was originally expected. The transient time of the switch is on the order of nanoseconds, thus a rise time on the order of a couple of microseconds would seem reasonable.

However, Fig. 4.3.2 shows that the rise time is about 1200us. This is longer than it would take to tune an ordinary LO from one frequency to another. The fall time for the RF carrier is measured to be 450us in Fig 4.3.3.

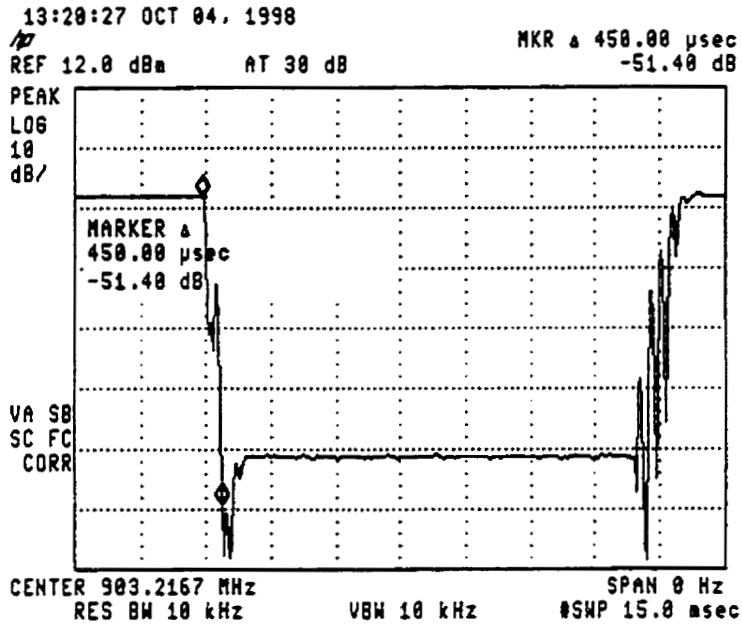



Fig. 4.3.3 Carrier Fall Time on 10ms Hops

Although, the fall time is better than the rise time it is still excessive. Attempts to use passive synchronization will fail with these long rise and fall times.

Next, the hopping interval is decreased to 1ms. Fig. 4.3.4 shows the performance of the agile synthesizer at a dwell time of 1 ms. As the channel dwell time is reduced the RF the ringing increases and makes the carrier unusable. In addition, for a hopping interval of 1ms the RF splatter has increased significantly.

The out of band energy interferes with adjacent channels and reduces the system capacity. There is approximately 45dB of power difference when the LO is on channel versus off-channel.

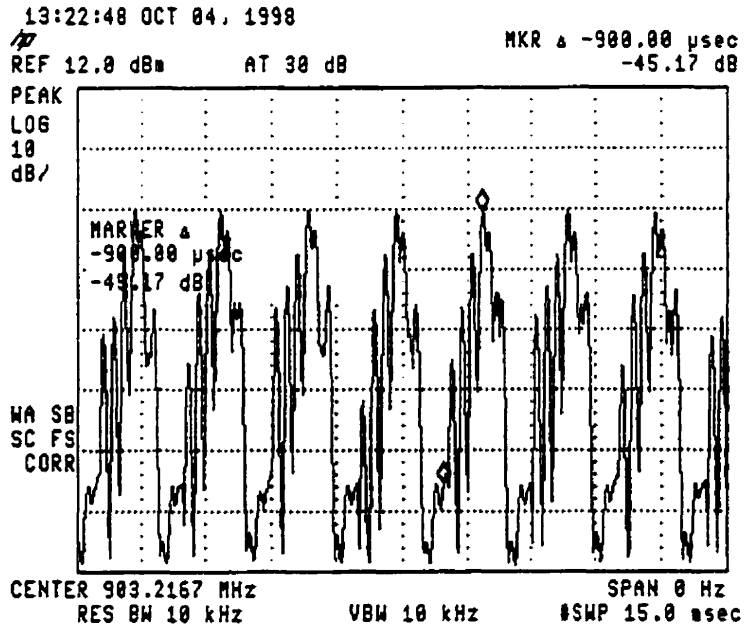



Fig. 4.3.4 Carrier Power on a Single Channel for 1ms Hops

The plot in Fig. 4.3.4. clearly shows that there is a serious problem with the design. An investigation began by looking for transients in each section of the agile synthesizer. Using a DSO and a spectrum analyzer each trace is probed for RF transients and voltage spikes.

After numerous tests the problem was traced to switched currents which were induced from the microcontroller as it toggled the control lines on the switch matrix. The switching currents were coupled through the power supply from the

microcontroller to the VCO. Measurements with the DSO showed that there is as much as 25mV spikes on the conditioned power rail feeding the VCO.

These low frequency transients are extremely difficult to filter since they require lowpass filters with very low cutoff frequencies. These filters are physically difficult to realize. At a hopping interval of 1ms the switch spikes have a frequency of 1kHz which is not filtered and enters the VCO power supply.

The term VCO pushing defines the sensitivity of the output frequency to variations in the voltage supplies. For the VCO used (Z580MC06) made by Z-communications the VCO pushing is 3MHz/V [19]. Voltage spikes of 25mV can cause as much as +/- 75KHz variation in the frequency. The RF ringing and long transient time are caused by these voltage spikes and the phase lock loop correction.

Next, modifications were made to the VCO by including separate linear power regulators to each of the VCOs. Isolation was also increased by replacing the long RF lines at the output of the VCO with RG316 coax cables. The same tests were run with a dwell time of 10ms. The output spectrum was captured and is shown in Fig. 4.3.5.

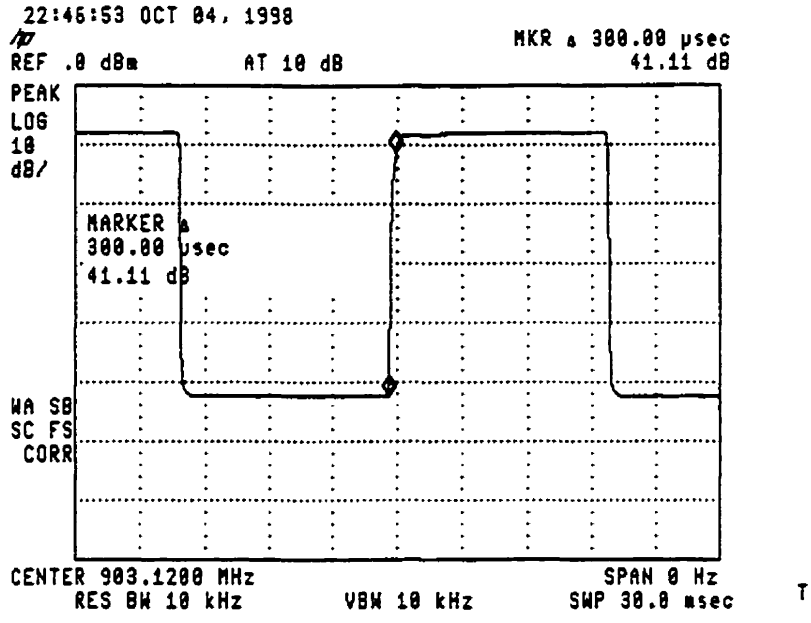



Fig. 4.3.5 Carrier Power on 10ms Hops

Fig. 4.3.5. shows the output spectrum of the agile synthesizer has significantly improved over the carrier power plot shown in Fig. 4.3.2. The transitions are smooth showing no ringing and less spurious energy as compared to the previous results.

Fig. 4.3.6 shows a closer look at carrier power vs. time plot with the sweep time set to 15ms. The markers show that the carrier is ramped up by about 45dB in under 40us. The video bandwidth of the spectrum is the post-detection filter which is set to 1MHz. The resolution bandwidth is also set to 3MHz, see Table 4-4 of HP Spectrum Analyzer User's Guide [22]. This will limit the rise time as seen in Fig 4.3.6.

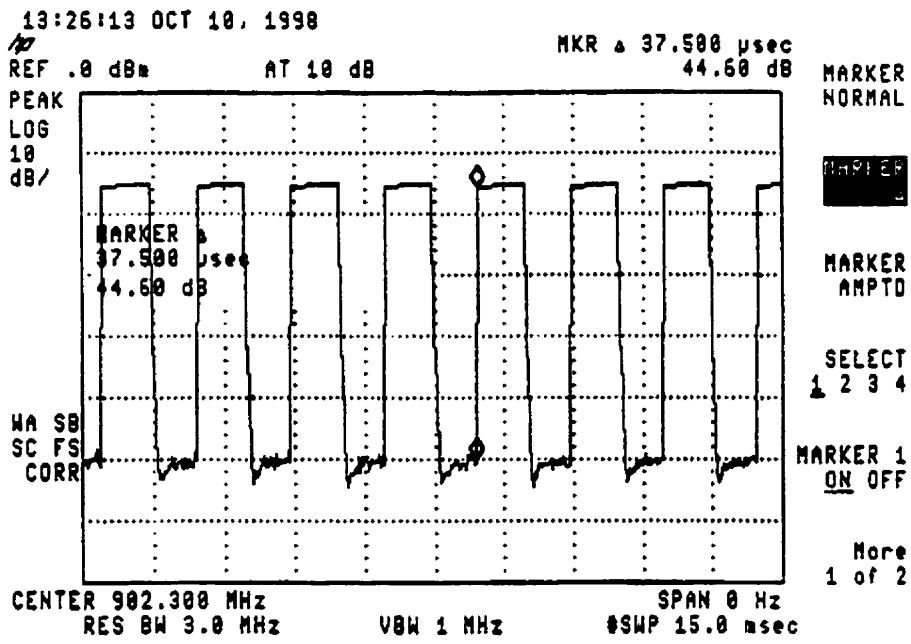



Fig. 4.3.6 Carrier Power on 1ms Hops

These tests have rescued the agile synthesizer. With careful power supply regulation and isolation the agile synthesizer works well and is practically realizable.

The next test focuses on spurious response of the agile synthesizer. The spectrum analyzer is setup on a frequency-magnitude sweep with a span of 1MHz and Hold Max activated. The firmware is modified so that hopping on the same channel is possible with 1ms dwell time. The output spectrum is shown in Fig. 4.3.7

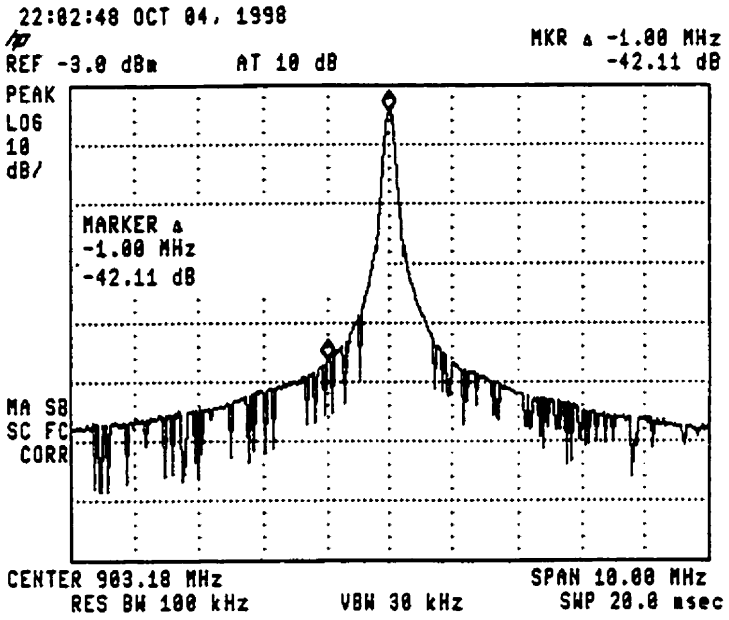



Fig. 4.3.7 Output Spectrum on Same Channel 1ms Hops at 10MHz Span

The agile synthesizer shows acceptable phase noise performance. The phase noise can be calculated the same way as before. The phase noise is -92dBc/Hz at 1MHz offset. This is the worst case since the HOLD MAX feature was activated on the spectrum analyzer. Using the averaging technique to 100 traces the phase noise for the agile synthesizer is 102dBc/Hz at 1MHz offset.

Fig. 4.3.8 shows the spectrum with a span of 1 MHz. The worst case phase noise is 74.8dBc/Hz at an offset of 100kHz. Once again averaging the phase noise gives an improvement of approximately 10dB. Then the typical phase noise is 85dBc/Hz at an offset of 100kHz.

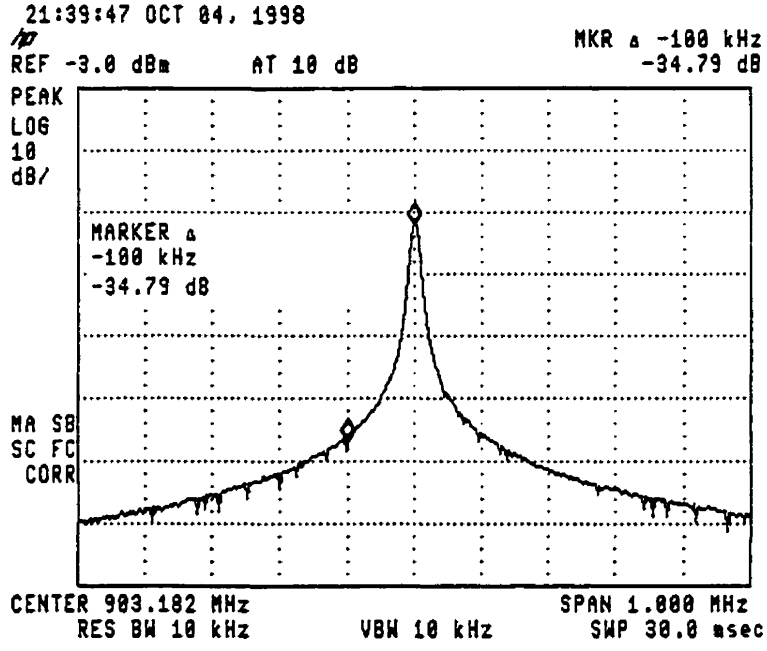



Fig. 4.3.8 Output Spectrum on Same Channel 1ms Hops at 1MHz Span

The agile synthesizer's phase noise is worse than that of a typical synthesizer. Typical phase noise performance is about -130dBc/Hz at 1MHz offset and -110dBc/Hz at 100kHz offset.

Next the agile synthesizer's acquisition time is compared to that of a typical synthesizer. A modulation domain analyzer shows a plot for a single PLL synthesizer (LMX1511) [21] see Fig. 4.3.9. For this synthesizer the loop reference frequency is 25kHz and the frequency jump is 58MHz. The acquisition time is approximately 1.5ms. Also note that the acquisition time depends on the magnitude of the frequency jump for the conventional synthesizer, but not for the agile synthesizer.

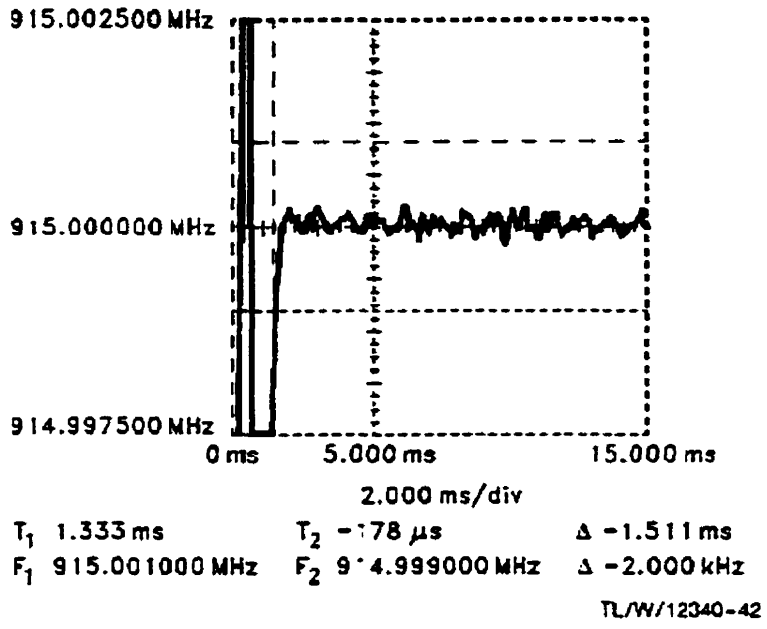



Fig. 4.3.9 A Modulation Domain Plot for a Conventional Synthesizer [21].

The agile synthesizer clearly has an advantage of quickly being able to switch between frequencies. With a tuning time of a few microseconds the agile LO is useful in rapid hopping applications. Table 4.3.1 compares the agile LO to a typical LO. In general the phase noise is increased by about 25dB for non-continuous phase agile synthesizer.

Table 4.3.1 A Typical Comparison of the Agile vs a Conventional Synthesizer.

| *Parameter* | *Conventional Synthesizer* | *Agile Synthesizer* |
|---|---|---|
| Phase Noise 100kHz offset dBc/Hz | -110 | -85 |
| Phase Noise 1 MHz offset dBc/Hz | -125 | -110 |
| Acquisition Time $\Delta f$ = 25MHz | 850us to 1500us | 40us |

## 4.4 Power Performance of the Agile Synthesizer

The major tradeoff with the agile synthesizer is the current requirement, since both local oscillators are required to run whereas the conventional synthesizer only uses one LO. This makes the agile synthesizer not practical for battery powered portable handheld communication devices. Table 4.6.1. shows the power budget for both the conventional synthesizer and the agile synthesizer. It is obvious that most of the additional current is used to power the second VCO. The power consumption of the agile synthesizer is almost double that of the conventional synthesizer.

Table 4.6.1 Current Consumption of the Agile vs Conventional Synthesizer.

| Component | Conventional Synthesizer | | Agile Synthesizer | |
|---|---|---|---|---|
| | Part | Power mW | Part | Power mW |
| PLL | LMX1511 | 18 | LM2335L | 20 |
| VCO | Single Z580MC06 | 85 | Dual Z580MC06 | 170 |
| Switch Matrix | None | 0 | FET Switches | 0 |
| Total | | 103 | | 190 |

In wireless LAN applications, where AC power is abundant the agile synthesizer may provide a practical solution for reducing the settling time in phase lock loops. In battery powered applications, lower power VCOs can be used inside the agile synthesizer with an amplifier at the output of the agile synthesizer.

## 4.5 System Performance

With its current implementation the agile synthesizer cannot be made to hop faster than about 1000 times per second. The limit is caused by the adjacent LO which must have enough time to acquire the next hopping frequency before it can be switched in.

As discussed in section 2.4 Fractional-N synthesis can be used to speed up the acquisition time for each individual LO. A more agile synthesizer could be designed using a dual Fractional-N PLL and the same switch matrix. In addition, further improvements to the acquisition time can be made using the Fast Lock feature of modern PLL synthesizers. The Fast Lock allows the cascading of a parallel filter which is activated only during acquisition. With these modifications a hopping rate of 2000 to 3000 hops per second may be achievable.

From a system perspective using an agile synthesizer improves latency, reduces overhead, allows for smaller packets, and provides better immunity to interference. However, it trades-off phase noise performance and power efficiency.

# CHAPTER FIVE

## 5.1 Conclusion

In this investigation an agile synthesizer is designed and tested. It was shown that the agile synthesizer is capable of superior lock times. Compared to the conventional synthesizer, the agile synthesizer has about one tenth the lock time and 25dB more phase noise.

The investigation only gave a brief description of the receiver implementation based on the agile synthesizer. Provisions were made to the agile synthesizer in order to accommodate its implementation with a dual receiver chain. This is left as an independent research project.

Frequency hopping modems can use the agile synthesizer core to provide much faster hopping, better throughput and lower latency. A brief summary of this investigation follows.

Chapter 1 discusses traditional frequency hopping spread spectrum radios. The basic components of a transceiver are outlined. Background information on phase lock loops, acquisition time, and fast frequency hopping is given.

Chapter 2 begins its focus with phase lock loops. The fundamental components of a phase lock loop are discussed; phase detector, voltage controlled oscillator, loop filter, and the frequency divider. Linear analysis, noise performance and the transient response of the PLL is also discussed.

Formulas for the design of the loop filter are derived. It was also noted that improved performance could be obtained by using Fractional-N synthesizer inside the agile synthesizer.
Chapter 2 gives background information on two types of RF switches; GaAs and PIN switches. GaAs switches prove to be more suitable for the implementation of the Switch matrix. This chapter concludes by comparing rapid frequency hopping and compares it to frequency chirps. It becomes evident that rapid frequency hopping is a staircase approximation to a frequency chirp with a low frequency acceleration factor.

Chapter 3 proposes a system design using the agile synthesizer in the transmit and the receive chain. This design can support simple FM modulation or a complicated vector modulation scheme. Matlab simulations compare the spectral output of coherent and non-coherent agile frequency synthesizers. Next focus is shifted from the system level design to the components of the agile synthesizer. A third order loop filter is designed and is used in both synthesizers. Schematics and gerber plots of the agile synthesizer are given. Finally, Chapter 3 concludes with the test setup.

Chapter 4 discusses the performance of the agile synthesizer. Both static and dynamic measurements of the switch matrix are made. The results show that the agile synthesizer is capable of an acquisition time of 40us compared to the conventional synthesizers whose acquisition time is on the order of a 1ms. However, it was found that the phase noise of the agile synthesizer is about 15dB worse than that of the conventional synthesizer. Another tradeoff is power consumption of the agile synthesizer is about double that of the conventional synthesizer.

## 5.2 Further Research

This investigation implemented the agile synthesizer for the transmitter. Future efforts can be made to implement the passive synchronization of a frequency hopping communication system. Although the agile synthesizer was designed with this capability, it was not tested here and is left for another investigation.

The agile synthesizer designed and tested in this investigation was non-continuous phase. More research needs to be done on continuous phase agile synthesizers. Such a device would have a much better phase noise performance than the non-continuous phase synthesizer.

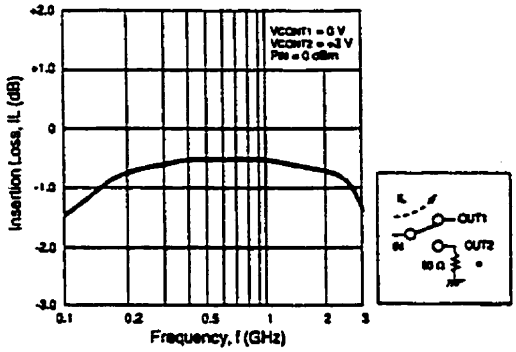
With a current push for wireless data networks, namely *Wireless Internet*, in the 2.4GHz and 5.8 GHz ISM bands a continuous phase agile synthesizer may offer an alternative to the direct sequence implementations. Currently, direct sequence has found its way into the IEEE802.11 standard and has been implemented into ASIC chipsets such as the Harris Prism™ chipset. A continuous phase agile synthesizer would remove some of the limitations of frequency hopping in high speed implementations and would offer its superior immunity to interference.

The ultimate goal of a continuous phase agile synthesizer is to migrate into the frequency chirp domain. This would provide a chirp communication system which could support very high speed communications of 100Mbps with robust multipath characteristics.

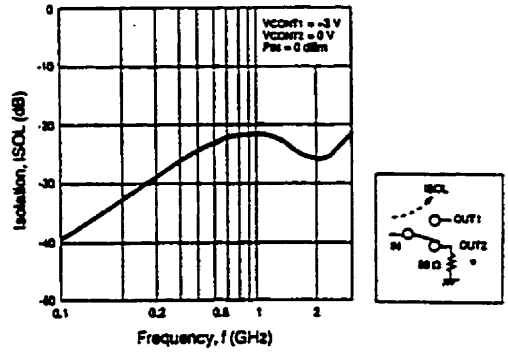
# APPENDIX A

## TYPICAL PERFORMANCE CURVES ($T_A$ = 25° C)

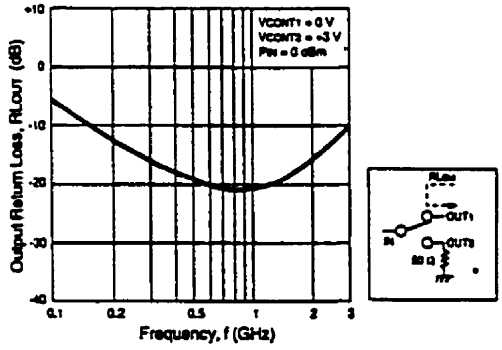
UPG132G
IN-OUT1 INSERTION LOSS
vs. FREQUENCY



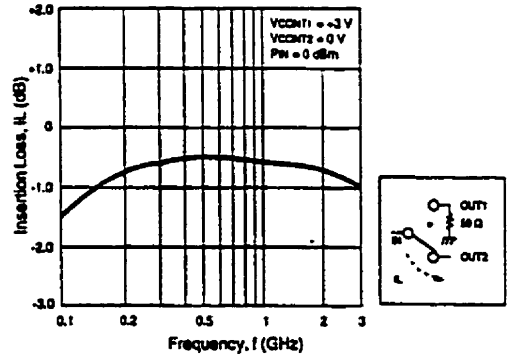

UPG132G
IN-OUT1 ISOLATION
vs. FREQUENCY



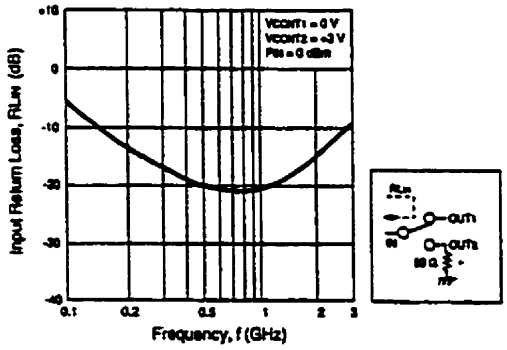

UPG132G
IN-OUT1 OUTPUT RETURN LOSS
vs. FREQUENCY



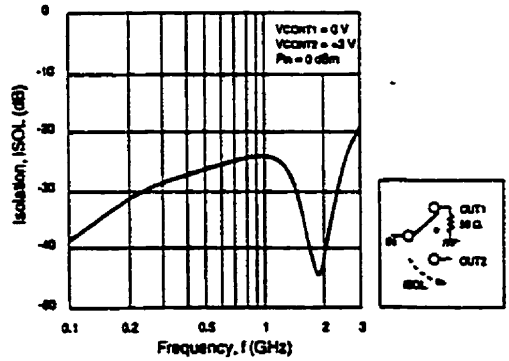

UPG132G
IN-OUT2 INSERTION LOSS
vs. FREQUENCY



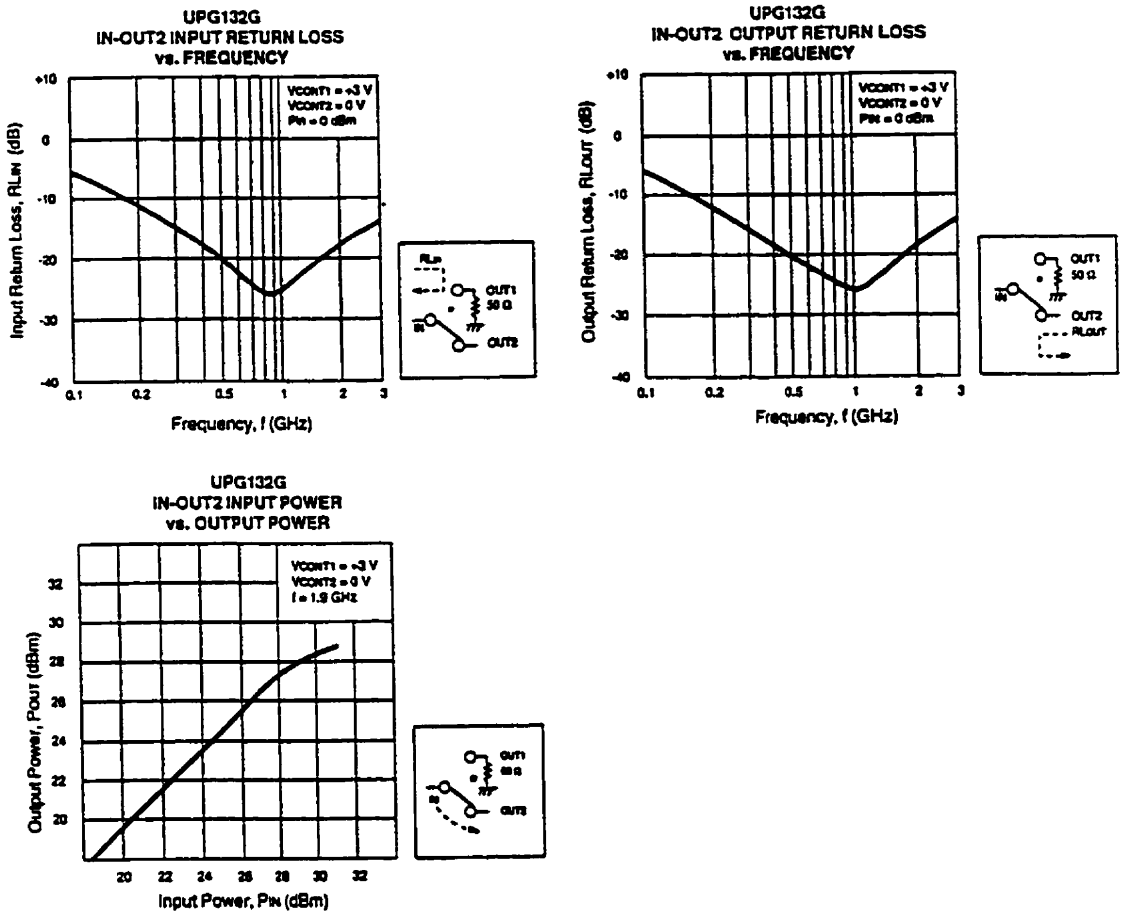

UPG132G
IN-OUT1 INPUT RETURN LOSS
vs. FREQUENCY




UPG132G
IN-OUT2 ISOLATION
vs. FREQUENCY




UPG132G Specifications taken from NEC datasheet [23].




UPG132G Specifications taken from NEC datasheet [23].

# APPENDIX B

```
// File:      DLS3.c
// Description:
//                This program drives the Dual synthesizer LMX2335TM.
//                It will be used to test the switch matrix.
//                Furthermore, it will be used to generate the Alige LO.
//
// Revisions:
//
//


#include <16C77.H>
#include <CTYPE.H>
#include <STDIO.H>

#use standard_io(A)
#use standard_io(B)
#use standard_io(C)

#fuses HS,NOWDT,NOPROTECT

#use delay(clock=20000000)
#use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7)

////////////////////////////
// Constants

#define LED_DELAY   100          // milliseconds
#define ESC 27

// Internal Hardware Profile

#define PIR1    0x0C
#define PIR2    0x0D
#define PortC   0x07
#define PIN_TX  2
#define CCP1IF  2

// External Hardware Profile
```

```
#define LE PIN_C0          // Latch Enable [LE_SYNTH or SYN LE active LOW]
#define CLK PIN_C3         // Must be low when enable is latched

#define LED1 PIN_D0
#define LED2 PIN_D0
#define LED3 PIN_D0
#define LED4 PIN_D0

#define LO1 PIN_D6
#define LO2 PIN_D5

#define MOD PIN_B3

#define TX1 PIN_B4
#define TX2 PIN_B0
#define RX1 PIN_B5
#define RX2 PIN_B1
#define LD PIN_D0

////////////////////////////
// Radio functions

// Radio Module state values
#define MR_IDLE 0
#define MR_TX   1
#define MR_RX   2

char    mr_state;          // radio module state


// ***************************************************************************
// Function:  mr_init
// Description:
// Set up the MR interface
// Turn on the synthesizer
// Clear the MR state
//
void mr_init()
{
        // disable SPI select
        output_high(LE);

        // disable Rx and Tx
```

```
        mr_state = MR_IDLE;
}

//*****************************************************************************


// *****************************************************************************
// Function:  main
// Description:
//      The main application code
//
void main()
{
        init();                 // Set up I/O

        printf("\r\nMR1 V002 ");      // Boot string
        printf(__DATE__);               // Compile Date


// Set up reference counter for the LMX2335

        // Fosc = 16MHz
        // High current output mode
        // Phase Detector Polarity Positive
        // Do normal operation mode
        // RF1 = 060282


        output_low(LE);
        spi_write(0x06);
        spi_write(0x02);
        spi_write(0x82);
        output_high(LE);


        // Fosc = 16MHz
        // High current output mode
        // Phase Detector Polarity Positive
        // Do normal operation mode
        // RF2 = 060280

        output_low(LE);
        spi_write(0x06);
        spi_write(0x02);
```

```
        spi_write(0x80);
        output_high(LE);


        // Set the output on Syn 1 = 902.3 MHz
        // Prescale is 64
        // Power Up mode
        // RF1 N 0118FF

        output_low(LE);
        spi_write(0x01);
        spi_write(0x18);
        spi_write(0xff);
        output_high(LE);


        printf("The Synthesizer 1 is ON 902.3");
        getc();

        // Set the output on Syn 2 = 902.4 MHz
        // Prescale is 64
        // Power Up mode
        // RF2 N 0118FD

        output_low(LE);
        spi_write(0x01);
        spi_write(0x19);
        spi_write(0x21);
        output_high(LE);


        printf("The Synthesizer 2 is ON 902.4");
        getc();

        output_low(RX1);          // RX1 is active high
        output_low(RX2);          // RX2 is active high
        output_high(TX1);         // TX1 is active high
        output_high(TX2);         // TX2 is active high

        printf("Unit in TX mode");
        getc();

loop:
```

```
// This transmitts LO1

        output_low(LO1);            // LO2 Active Low
        output_high(LO2);           // LO1 Active High


//      printf("Transmitting LO1");
//      getc();

        delay_ms(10);


// This transmitts LO2

        output_low(LO2);            // LO1 Active Low
        output_high(LO1);           // LO2 Active High


//      printf("Transmitting LO2");
//      getc();

        delay_ms(10);
        goto loop;

//      }


}
// ***************************************************************************
```

# REFERENCES


[1] Bernnan, P. V., "Phase Locked Loops: Principles and Practice", Macmillan, 1996

[2] Microhard Systems, "TRX-900 Spread Spectrum Radio DataSheet", Microhard Systems Inc., 1997

[3] National Semiconductor, "National Products for Wireless Communications DataBook: LM2320TM Application Note", National Semiconductor Corporation, 1997.

[4] John McRory, "CAD for Linear and Nonlinear RF/Microwave Circuit Design", University of Calgary, Department of Electrical and Computer Engineering, 1996.

[5] Best, Roland E., "Phase-Locked Loops Theory, Design, and Appications", McGraw-Hill,1984.

[6] Rhea, Randall W., "Oscillator Design & Computer Simulation", 2nd ed., McGraw-Hill, 1995.

[7] Robert Silverstein, "Design Techniques Enhance Isolation In Switch Assemblies" , Microwaves&RF December 1997

[8] Rohde, Ulrich L., "Digital PLL Frequency Synthesizers Theory and Design", Prentice-Hall, 1983.

[9] Egan, W.F., "Frequency Synthesis by Phase Lock", John Wiley & Sons, 1981.

[10] Best, Roland E., "Phase-Locked Loops Theory, Design, and Applications", 2nd ed., McGraw-Hill Inc. 1993.

[11] Gardner, F.M., "Charge-Pump Phase-Lock Loops", IEEE Trans. Commun., vol. COM-28, pp 1849-1858, Nov 1980.

[12] Nathanson, Fred E., "Radar Design and Principles", 2nd ed., McGraw-Hill, 1991.

[13] Hewlett Packard, "Voltage Controlled Oscillators Evaluated for System Design", Application Note M024, Communications Components Designer's Catalog, 1997

[14] Philips Semiconductor, "Low Power Single/Dual Frequency Synthesizers: UMA1017M/1018M/1019m(AM)/1020M(AM)", Application Note AN95102, Application Handbook, IC17b, 1998

[15] T. Saba, D. K. Park, and S. Mori, "Fast-Acquisition PLL Synthesizer Using Parallel N-Stage Cycle Swallower with Low Power Consumption and Low Phase Noise," IEEE Transactions on Vehicular Technology, vol. 44, May 1995, pp.296-302.

[16] Hewlett Packard, "A Low-Cost Surface Mount PIN Diode π", Application Note 1048, Communications Components Designer's Catalog, 1997

[17] Hewlett Packard, "A Low Distortion PIN Diode Switch Using Surface Mount Devices", Application Note 1049, Communications Components Designer's Catalog, 1997

[18] Mini-Circuits, "Switchs - Modern Definition of Terms", Chapter 13, RF/IF Designer's, 1992

[19] Zcomm VCOs Datasheets for V580C06

[20] National Semiconductor, "National Products for Wireless Communications DataBook: LMX2335TM Datasheet", National Semiconductor Corporation, 1998.

[21] National Semiconductor, "National Products for Wireless Communications DataBook: LMX1511TM Datasheet", National Semiconductor Corporation, 1998.

[22] Hewlett Packard, "HP 8590 E-Series and L-Series Spectrum Analyzers User's Guide", 1998

[23] California Eastern Laboratories, "L-Band SPDT MMIC Switch - UPG132G", RF and Microwave Semiconductors, 1998

[24] Proakis, John G., "Digital Communications", 3rd ed., McGraw-Hill 1995.