Phase Locked Loop Design for Transmitting and Receiving Sections in Optical Wireless Access

Ampornrat Posri
1075007

A dissertation submitted to
Kochi University of Technology
in partial fulfillment of the requirements for
The master degree of engineering

Supervisor
Associate Professor Masayoshi Tachibana

Course of Electronic and Photonic Engineering System
Kochi University of Technology
Kochi, Japan

February 2005
Acknowledgement

This research was achieved not only through the efforts of mine. The contribution of advice or consultation greatly contributes to the advancement of the research. It is necessary to acknowledge those whom have contributed kindly to the research. The author would like to first thank Kochi University of Technology for allowing the establishment of this research. Especially thank to my advisor, Associate Professor Masayoshi Tachibana for his valuable advice, guidance and support and associate professor Koji Nonaka for his advice on the concept of system level of the research. Their advice has inspired me to work efficiently.
Abstract

The design of transmitting and receiving parts of optical wireless LAN system by using phase locked loop circuit is presented. With the optical wireless access, single beam communication is required. The Hub-beam direction control signal should be transmitted simultaneously with the data signal. Phase locked loop circuit is utilized to demonstrated as frequency synthesizer for transmitting section and as clock recovery in receiving section. The circuits of both parts are implemented on the circuit board.
List of Figure

Fig 1.1.1 Basic operation of optical wireless access system
Fig 1.3.1 Research framework
Fig 2.1.1 Block diagram of phase locked loop
Fig 2.1.2 Simplified block diagram of PLL with individual transfer functions
Figure 2.2.1 XOR phase detector
Figure 2.2.2 Phase frequency detector
Fig 2.4.4 Passive low pass filter
Fig 2.4.5 Active low pass filter
Fig 2.4.6 Charge-pump loop filter
Fig 2.4.7 VCO characteristic
Fig 2.3.1 Phase locked loop with frequency divider
Fig 2.5.2.1 Phase noise model for PLL
Fig 3.1.1 System level of transmitting section
Fig 3.1.2 System level of receiving section
Fig 3.2.1 Phase locked loop 1st model
Fig 3.2.2 Simulation result of low pass filter of 1st model
Fig 3.2.3 PLL model with adding 2nd order LPF
Fig 3.2.4 Simulation result of PLL model in Fig 3.2.3 (narrow deviation range)
Fig 3.2.5 Simulation result of PLL model in Fig 3.2.3 (wide deviation range)
Fig 3.2.6 PLL model with using PFD and active LPF
Fig 3.2.7 Simulation result of model in Fig 3.2.6
Fig 3.2.8 Power spectrum of two different synthesized frequencies
Fig 3.2.9 PLL model with added data signal and FIFO
Fig 3.2.10 Power spectrum of transmitting NRZ signal
Fig 3.3.1 Clock recovery model by PLL
Fig 3.3.2 Control signal in clock recovery 1st model
Fig 3.3.3 Clock recovery model with changing PD
Fig 3.3.4 Output signal of LPF of model in Fig 3.12
Fig 3.3.5 Power spectrum of two different recovered signals
Fig 4.1.1 Frequency divider design flowchart
Fig 4.1.2 Symbol of frequency divider
Fig 4.1.3 simulation result of frequency divider No.1
Fig 4.1.4 Simulation result of frequency divider No.2
Fig 4.1.5 Simulation result of frequency divider No.3
Fig 4.1.6 Symbol of phase frequency detector
Fig 4.1.7 Simulation results of PFD (f_{ref_{signal}} < f_{div_{signal}})
Fig 4.1.8 Simulation results of PFD (f_{ref_{signal}} > f_{div_{signal}})
Fig 4.1.9 (a) top-level symbol of design (b) sub-level of design
Fig 4.3.1 Design placement and routing process
Fig 4.3.2 Physical layout of the design
Fig 5.1.1 Graph of experiment values of oscillation frequency of VCO versus tuning voltage
Fig 5.1.2 Circuit diagram for shifting dc level
Fig 5.1.3 Low pass filter circuit
Fig 5.1.4 Difference amplifier circuit
Fig 5.1.5 Circuit implementation of transmitting section
Figure 5.16 Circuit Schematic of transmitting section
Fig 5.1.7 Experiment for transmitting system operation
Fig 5.1.8 Spectrum of 100 MHz frequency synthesis (used cutoff value from simulation)
Fig 5.1.9 Phase noise measuring
Fig 5.1.10 Power spectrum of 100 MHz frequency synthesis
Fig 5.1.11 Power spectrum of 100.1 MHz frequency synthesis
Fig 5.2.1 Non-inverting summing amplifier circuit
Fig 5.2.2 PLL circuit diagram of receiving section
Fig 5.2.3 Experiment of receiving system operation
Fig 5.2.4 Power spectrum of 100 MHz recovered signal
Fig 5.2.5 Power spectrum of 100.1 MHz recovered signal
List of Tables

Table 3.2.1 Adjusting parameters in PLL model with simulation results
Table 4.4.1 pin assignment of the design to FPGA chip
Table 5.1.1 Experiment values of oscillation frequency of VCO versus the tuning voltage
# Table of Contents

List of Figures iii  
List of Tables v  
Chapter 1 Introduction  
1.1 Introduction 1  
1.2 Research Objective 1  
1.3 Research framework 3  
Chapter 2 Literature Reviews  
2.1 Fundamental of phase locked loop 4  
2.2 Basic component of PLL 5  
2.3 Frequency Synthesizer by Phase Locked Loop Circuit 10  
2.4 Clock recovery 11  
Chapter 3 System Level Design  
3.1 System level design for Signal transmitting and Signal Receiving 13  
3.2 Phase Locked Loop Model in transmitting Section 14  
3.3 Phase Locked Loop Model in receiving Section 22  
Chapter 4 LSI Design Process  
4.1 VHDL Coding and simulation results 26  
4.2 Design Synthesis 31  
4.3 Design Layout Process 31  
4.4 Design Implementation to FPGA 33  
Chapter 5 Circuits Implementation and System Operation  
5.1 Circuit Implementation of Transmitting Section 34  
5.2 Circuit Implementation of Receiving Section 42  
Chapter 6 Conclusion  
Conclusion 47  
Future Works 48  
References 49  
Appendix A
1.1 Introduction

In the past few decades the great advancement in information technology has become an important role due to the essential part of home and business world of information. The use of wireless has grown rapidly. The advantages of wireless are rapid deployment, without the need to dig trenches for cables, and seek permissions for right of way and allow people to communicate while they are mobile. Optical wireless system is the ones of advanced information technology that has the several advantages over the other technology. Optical wireless system is similar to a fixed wireless system except that data is carried by optical or infrared beam rather than microwave or radio frequency carrier. Optical wireless communications have certain advantages in comparison with both RF and optical fiber systems. They can deliver in wide bandwidths and also provide free-space communications comparable to RF links.

Optical wireless LAN is the system in which every computer can communicate with other systems. This system is becoming increasingly common in small offices where installing wired network has too much trouble [1]. Assume that the optical wireless LAN system consists of one main server system (upper plate) and the end node (client computers) as shown in Fig1.1.1. Signal communication between server and end node operates via the optical or infrared beam by using the Laser Diode (LD) as the optical transmitter and Photo Detector (PD) as optical receiver. The transmitting part consists of the subsystem of the laser diodes which each laser diode has it own specific channel for signal transmitting. From this Figure, suppose the communication occurs in right side channel, while the end node is moving toward another coverage area. The single data beam from end node which includes data signal and control signal is sent to the hub node. The control signal will be separated from received signal at the server site and then will be used to control the switching matrix in order to transmit the signal in the appropriate channel for the new location of the end node. The switching matrix system utilizes the control signal to tracking the signal between server and end node. At this point the communication changed to the middle channel [2].

1.2 Research Objective

With single beam communication data signal and control signal will be transmitted simultaneously from transmitting section while the combined signal will be separated at receiving section. In this system the information is transmitted in high data rates while the control data signal has low data rates. The purposes of research are combining two signals together in transmitting section and separating these signals at the receiving section without converting the data frame format.
Fig 1.1.1 Basic operation of optical wireless access system
1.3 Research Framework

My research framework was started with the system level design by using Matlab software. Next step was the LSI design process for digital parts and after with the analog parts design process. Finally all components were implemented in the circuit board to operate the whole system as illustrated in Fig 1.3.1.
Chapter 2
Literature Reviews

2.1 Fundamental of Phase Locked Loop (PLL)

Phase locked loop is a circuit which synchronizes (locked state) the frequency of the output signal generated by an oscillator with the reference or input signal in frequency or phase. The differences between the reference signal and output signal is called phase error and the control mechanism acts on the oscillator in such away to reduce the phase error. The phase of the output signal is actually locked to the phase of the reference signal [3].

Phase-Locked Loop, PLL, is widely used among the electronics and communication systems, for clock and data recovery, frequency synthesis, clock synchronization in microprocessors, and many applications [4].

![Fig 2.1.1 Block diagram of phase locked loop](image)

Phase locked loop circuit consists of basic components as below
1. Reference signal generator (RG)
2. Phase detector (PD)
3. Low pass filter (LPF)
4. Voltage controlled oscillator (VCO)

Refer to Fig 2.1.1[5]; reference signal generator is generated by oscillators which typically use crystal oscillator. Phase detector compares the phase of reference signal and output signal which generated from VCO. The output of phase detector is proportional to their phase difference. The phase detector gain is expressed in volt of output per radian of phase difference. After the signal passed the phase detector, it is necessary to provide a low pass filter in order to reject the carrier frequency component and high frequency noise. Voltage controlled oscillator (VCO) oscillates at instant angular frequency which is determined by the output from loop filter.

To get insight into the function of PLL, Fig 2.1.2 introduces the Laplace transfer functions of individual building circuits for investigation of small signal properties.[3]
Refer to Fig 2.1.2, phase detector compares input phase $\phi_i(t)$ with the output phase $\phi_o(t)$ and the output $v_d(t)$ proportional to the phase difference of respective input signals. The factor $K_d$ [V/rad] called phase detector gain.

$$v_d(t) = [\phi_i(t) - \phi_o(t)] K_d$$

The output signal from phase detector, $v_d(t)$ passed the loop filter, $F(s)$ which used to attenuate all undesired sidebands and high frequency noise. The DC output voltage $v_2(t)$ is produced by the convolution between time response of low pass filter and phase detector output as shown below.

$$v_2(t) = v_d(t) \otimes h_f(t)$$

The output $v_2(t)$ is applied to the frequency control element of VCO. The output phase is

$$\phi_o(t) = \int \omega_o(t)dt = \omega_o(t) + \int K_o v_2(t)dt$$

where $\omega_o$ is the VCO free-running frequency. The factor $K_o$ [2πHz/V] is the oscillator gain. The oscillated frequency from VCO is determined by the output voltage from loop filter in the way to reduce the phase error.

## 2.2 Basic Components of PLL

**Phase Detector**

Phase detector has two frequency inputs and produces the output voltage which proportional to the phase difference between two inputs. There are two form of phase detector which is analog PD and digital PD. Basically analog PD perform as mixers with an IF centered at zero frequency. There is the limitation for using this type of PD which that the output is proportional to the sine of phase difference rather than to the phase difference. Digital PDs are based on applications of simple flip-flop gates or more complicated combinations.

**Exclusive-OR Phase Detector**

Exclusive OR PD has the advantages due to their simplicity and compatibility with IC design. The operation of XOR as PD is shown in Fig 2.2.1. The PD characteristic
is triangular waveform which phase difference ranges from $\frac{\pi}{2}$ to $\frac{3\pi}{2}$. However, there is the disadvantage of using XOR PD. Suppose the duty cycle of reference input is 0.5 but that of the output signal from PLL is not 0.5, it produce a nonzero free running voltage which is defined as PD average output when there is no input Vi and also reduce the maximum voltage of PD characteristic[6].

![XOR phase detector](image)

**Figure 2.2.1 XOR phase detector**
(a) XOR symbol  (b) XOR operation  (c) output characteristic
Two-State Phase Detector

R-S and J-K Flip-Flop can be used as Phase Detector. By using the inputs signal to the Set input and the other to reset input pin. The Q and NQ output a rectangular waveform. The average value of the output signal is proportional to the phase difference between two input signals. The DC output is a saw-tooth wave which the phase difference ranges from 0 to $2\pi$ or $-\pi$ to $+\pi$. These PDs have the advantages due to their simplicity and compatibility with IC design. Another benefit is the increase in linear range, double that of XOR PD and the duty cycle of two input signal are not important. One disadvantage of this circuit is that cause a lot of spurious reference harmonics.

Phase Frequency Detector (PFD)

These PDs are often used in practice due to the acceleration of phase lock. This is because these PDs combined the properties of frequency and phase detecting together [3]. One example of phase frequency detector is shown in the Fig 2.2.2 (a). The operation of PFD can be explained by the timing diagram in Fig 2.2.2(b). Assume frequency $f_1$ is higher than $f_2$, only output $V_1$ is activated. Oppositely, if $f_2$ is higher than $f_1$, only output $V_2$ is activated. The characteristic graph is saw-tooth waveform with the operation range 0 to $2\pi$ [3].
Loop Filters

For 1st order PLL that has no loop filter included, just has one degree freedom which is DC gain that can be used for changing the corner frequency of simple PLL [3]. The stop band of the PLL characteristic is only -20 dB/dec. In some case we need to increase the attenuation of PLL for high frequencies, the part of filter will be included to PLL to satisfy more complex specification. This filter acts as attenuator at high frequency which has unity gain at DC.

Passive Filter

In simple case, a passive RC network can be used in PLL. Examples of passive low pass filter are shown below.

A simple RC Filter

Simple RC filter circuit is illustrated in Fig 2.4.4 (a) with transfer function is

\[ F(S) = \frac{1}{1 + sRC} \quad F(S) = \frac{1}{1 + s\tau} . \]

Phase Lag-Lead RRC or RCC Filter

These circuits shown in Fig 2.4.4(b),(c) has transfer function equal to

\[ F(S) = \frac{1 + S\tau_2}{1 + S\tau_1} . \]

For circuit (b), \( \tau_1 = C(R_1 + R_2) \) and \( \tau_2 = CR_2 \)
For circuit (c), \( \tau_1 = (C_1 + C_2)R \) and \( \tau_2 = C_2R \)
Active Filter

Although, the passive filter can be used to reject the carrier and high frequency components, but its maximum DC gain is unity. The static phase error will then not be zero and the loop bandwidth will be coupled with the static phase error [7]. In order to reduce the static phase error, a large DC gain of loop filter is desirable [6]. Active filter satisfy this requirement due to its infinite gain.

The examples of active filter used in PLL are illustrated in Fig 2.4.5. (a) and (b) with their transfer function [3], [8].

![Active low pass filter](image)

(a) \[ F(s) = \frac{1}{s^2 R^2 C_1 C_2 + 2sRC + 1} \text{, cutoff frequency} = \frac{1}{\sqrt{R^2 C_1 C_2}} \text{ rad/s} \]

(b) \[ F(s) = \frac{1 + R_2 C_1 s}{R_1 C_2 s} \]

The alternative to op-amp loop filter is the use of a charge pump working with RC network [7]. As shows in Fig 2.4.6, phase detector is assumed to provide a digital “pump up” or “pump down” signal. If VCO output is lagging the input reference, output signal from PD will activate top current source, depositing charge onto capacitor. If VCO is leading, the bottom is activated, drawing the charge from capacitor [8].

![Charge-pump loop filter](image)

Voltage Controlled Oscillator (VCO)

Fig 2.4.7 shows typical VCO Characteristic. The VCO frequency \( \omega_0 \) is a linear function of the control voltage \( v_c \) as \( v_c \) varies from 0 to 4 V, the VCO output varies over the range of 8 Mrad/s to 16 Mrad/s. When PLL is in the lock state with \( \omega_0 = \omega_i \) from the characteristic graph says \( \omega_i = 10 \text{ Mrad/s} \). It requires \( v_c = 1 \text{ V} \). The value of \( v_c \) at \( \omega_i \) is called the static control voltage \( V_{CO} \). It is convenient to refer to the output frequency
deviation $\Delta \omega_i \equiv \omega_0 - \omega_i$. At the lock state, average of $\omega_0$ equals $\omega_i$ so the value of $\Delta \omega_i$ is a measure of how far $\omega_0$ is from its average in lock. The slope of the VCO characteristic is called the VCO gain $K_0$ which the unit is rad/s/V or Hz/V [6]. The output phase of VCO cannot be determined only from the present value of control voltage. It depends on the history of control voltage also [4].

![Fig 2.4.7 VCO characteristic](image)

The following parameters are important when VCO be used in PLL. 1) Tuning range which the variation of output amplitude and jitter must be minimal. 2) Jitter and phase noise: timing accuracy and spectral purity requirements impose an upper bound on VCO jitter and phase noise. 3) Supply and substrate noise rejection: VCOs must be highly immune to supply and substrate noise. 4) Input/Output characteristic linearity: If PLL used as FM demodulator variation of $K_{VCO}$ introduces harmonic distortion in the detected signal and must be below 1%.

2.3 Frequency Synthesizer by Phase Locked Loop Circuit

Frequency synthesizer produces accurate frequencies of reference frequency or multiples of the reference frequency. In this case a frequency divider is included in the feedback path of PLL as shown in Fig 2.3.1. Frequency Divider (N) is used to divide the output signal from VCO N times, other way to say is that the output of the VCO generates frequency which is N times of the reference signal. In practice, integer-N, fractional-N and higher order fractional-N are used. The integer N can be varied and selected by user.

![Fig 2.3.1 Phase locked loop with frequency divider](image)
2.3.1 Fractional-N PLL frequency synthesizer

Its function similar to the divider-by-N PLL but the output frequency of the VCO is not limited to the multiples of reference signal only but can be locked to fractional multiples of reference frequency [4].

2.4 Clock Recovery

Many systems such as digital communication system or optical communications, the data are transmitted with no accompanying clock. Almost communication system, information is sent by a series of bit 1’s and 0’s with Non Return to Zero (NRZ) format. Clock recovery utilizes phase lock property of PLL to recovery the clock signal from the binary data stream or synchronizes a clock to the data. The application of PLLs to clock recovery has some special design consideration due to the random nature of data especially the phase detector design [4].

2.5 Noise and Jitter

2.5.1 Types of noise

All physical are subjected to some sort of uncertainties due to fluctuations of individual internal or external parameters. If the respective changes are small and random, we called it “noise”. In practice, we encounter three fundamental types of noise that differ by the power in the time or frequency unit $S(f)$, that being called the power spectral density (PSD)[3].

The three types of noise are as follows

**White Noise**

Typical representative of white noise is the black body radiation and thermal noise of resistors or shot noise in electronic devices. White noise composes of two types.
- Thermal noise
- Shot noise

**Flicker or 1/f Noise**

The flicker noise was found that at very low frequencies the magnitude of the excess spectral density varied as the current square, and it was frequency dependent. The same type of noise was observed with India ink resistors, carbon microphones, and many other electronic devices. The PSD law is $S_n(f) \sim 1/f^\alpha$, where the power of $\alpha$ was vicinity of 1.

**Noise 1/f^2**

The PSD of this type of noise is $S(f) \sim 1/f^2$ and it is closely connected with the Brownian motion. The first observations were of mechanical particles. However, later studied proved that many thermal, electric, and electronic processes might be solved with the same mathematical approach.
2.5.2 Phase Noise in PLL

In PLL model, VCO changes an error voltage to frequency and then amplitude modulated noise present in the loop after phase detector convert to phase term. This can occur at any point in PLL. For frequency synthesizer, AM noise also converted to phase modulation at digital counter which cause the output to have timing jitter. This is pass trough phase detector and appears on the output of VCO [9]. Fig 2.5.2.1 illustrates the additive phase noise sources in PLL model.

![Phase noise model for PLL](image)

At reference generator, phase noise \( \phi_r(f) \) is introduced associated with its output. Amplifier in loop filter generates the amplitude noise which will be subsequently converted to phase modulation, \( \phi_l(f) \). VCO itself generates phase represented by \( \phi_v(f) \). Frequency divider also introduces phase instabilities \( \phi_d(f) \) in the loop.
Chapter 3
System Level Design

3.1 System level design for Signal transmitting and Signal Receiving

In order to combine the low speed control signal with the high-speed data in single beam communication, Phase Locked Loop circuit was utilized to function as frequency synthesizer, generate the clock signal for the data buffer in the transmitting part. It was also operated as clock recovery in the receiving part to retrieve the clock signal. Process of signal transmitting and signal receiving are explained below.

3.1.1 Transmitting Section

Refer to Fig 3.1.1, a low speed control signal, in 10 kbps are combined with the high speed data signal in 100Mbps, the PLL circuit is designed as frequency synthesizer, which used to generate two different frequencies dependent to control signal, from this model when control bit ‘0’ PLL circuit will generate 100 MHz, when control bit ‘1’ the circuit will generate the output signal at 100.1MHz. After that the synthesized signal is used as the clock signal of data buffer or First-In First-Out (FIFO) data buffer to drive out the data signal. At this step, the control signal is merged with the data signal already.

3.1.2 Receiving Section

PLL is designed as clock recovery circuit; this is because the received signal is NRZ pattern with no accompanying clock signal. As illustrated in Fig 3.1.2 the received signal is injected to PLL circuit, at the steady state of PLL system, the clock signal will be recovered from received signal, which are 100 MHz or 100.1MHz depends on the control signal in the transmitting part. From this point, the recovered clock signal is used for retiming the received signal to retrieve the data signal. The clock signal will multiply with 100MHz signal. At this step, the results are 0 Hz and 200 MHz signals for control bit “0” time period and 100 kHz and 200.1 MHz for control bit “1” time period. Low pass filter is used to reject the high frequencies of this signal and then the control signal can be recovered.
3.2 Phase Locked Loop Model in Transmitting Section

As explained in chapter 1, the phase locked loop circuit has to be designed to synthesize two different frequencies, which equal to 100MHz and 100.1 MHz. Basic components of phase locked loop circuit was explained in chapter 2.

The PLL model was modeled by using MATLAB Software in application of SIMULINK. Each component inside the model was changed and simulated in order to find the optimal result. The first model showed in Fig 3.2.1 used exclusive-or (XOR) gate to work as phase detector due to its simplicity. Simple RC low pass filter which transfer function is

\[ F(S) = \frac{1}{1 + sRC} \]; cut off frequency = \( \frac{1}{RC} \)

The “Convert to square wave” block is used to convert the sine wave that output from VCO to be square wave before injected to frequency divider. The “Accumulator” block is sub-block used to control frequency divider block. Refer to the requirement for
fractional divider that explained above, the accumulator block was modeled to generate “Carry” signal to equal “1” in every 10 clock of reference signal for the control signal equal to “1”. The “Carry” signal is used to control the frequency divider block to divide the output signal from PLL by 101. All parameters of the model, which are cutoff frequency of low pass filter, the oscillation frequency of VCO and sensitivity gain of VCO were adjusted until the output signal reached the specification.

Fig 3.2.1 Phase locked loop 1st model

Fig 3.2.2 Simulation result of low pass filter of 1st model
The simulation result of the model in Fig 3.2.2 shows the VCO control signal (output signal of LPF). The graph is the voltage of output of LPF versus the simulated time that operated within 0.3 ms (3 bit of control signal consists of bit 1 0 1 as shown in Fig 3.2). The graph shows the two different levels of voltage which means two different frequencies were synthesized for the model. The ranges of frequency in each of control bit were very high and overlapped each other which caused the problems in receiving section for separating these two different output signals. The high overshoot in the first bit of control data was ignored because the first bit was determined as initial bit.

The cause of problem was assumed that it may occur from the type of LPF used therefore the model was changed the type of LPF to

\[
F(S) = \frac{1+S\tau_2}{1+S\tau_1}.
\]

The simulation result after changing LPF gave the better results, locked ranges narrower for two output frequencies but the range still high and may affect the process in receiving section. This model was modified again by adding this filter to the model in Fig 3.2.1 to reduce the high frequency from first filter as shown in Fig 3.2.3

![Fig 3.2.3 PLL model with adding 2nd order LPF](image)

Adjusting the parameters in the model affected the simulation results. If cutoff frequency of filter is decreased, the frequency deviation range which is the frequency range of the output of PLL in stable state will decrease but the settling time of system will increase (settling time is the time used before the system reach the stable state). If increases the sensitivity gain of VCO the settling time will decrease but it causes the high
lock in range. Settling time and frequency deviation range of output signal has to be trade off when assigning output requirement. Therefore the parameters in PLL circuit have to be adjusted in the way to give optimum results. By comparison the graphs in Fig 3.2.4 and Fig 3.2.5, the narrower deviation range, the higher settling time.

![Fig 3.2.4 Simulation result of PLL model in Fig 3.2.3 (narrow deviation range)](image)

Fig 3.2.4 Simulation result of PLL model in Fig 3.2.3 (narrow deviation range)
Many types of phase detector described in chapter2 were used in PLL to observe the behavior of output signal and find the optimal type. Fig 3.2.6 shows the modified PLL model by changing phase detector part from XOR to phase frequency detector. The control graph of this model is shown in Fig 3.2.7.

The passive filter used in the first filter block was changed to the active filter to make it more practical (the advantage of using active filter was explained in chapter2). Transfer function of active low pass filter is

\[
F(s) = \frac{1}{1 + 2S\tau_2 + S^2\tau_1\tau_2}; \text{ cut off frequency } = \frac{1}{\sqrt{\tau_1\tau_2}} \text{ rad/s}
\]

Table 3.2.1 shows parameters adjustment with the simulations result (settling time and deviation range) of simulation of model in Fig 3.2.6. The control graph in Fig 3.2.7 shows the settling of the output signal of 100MHz equal to 0.0225 ms and 0.0205 ms for output signal of 100.1 MHz. The settling times are within 25% of control signal period (0.1 ms). The PLL circuit generates output signal with frequency in locked range within 0.025 ms therefore there is 0.075 ms left for recovering the signal in receiving part. Suppose the recovery circuit recover the clock signal within 0.025 ms after the PLL synthesize the output signal in transmitting part, so the control signal will be retrieved within 0.05 ms. 0.05 ms time left for one bit control signal doing its function. The results in table 3.2.1 show that using cutoff frequency around 800 krad/s, the acceptable outputs will be received.
Fig 3.2.6 PLL model with using PFD and active LPF

Table 3.2.1 Adjusting parameters in PLL model with simulation results

<table>
<thead>
<tr>
<th>2nd Active Low pass filter</th>
<th>simple RC Filter</th>
<th>Output signal of 100MHz for control data bit ' 0 '</th>
<th>Output signal of 100.1MHz for control data bit ' 1 '</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1/\tau_1$ (krad/s)</td>
<td>$1/\tau_2$ (krad/s)</td>
<td>Cut off frequency (krad/s)</td>
<td>Cut off frequency (krad/s)</td>
</tr>
<tr>
<td>1000</td>
<td>1000</td>
<td>1000</td>
<td>950</td>
</tr>
<tr>
<td>900</td>
<td>900</td>
<td>900</td>
<td>950</td>
</tr>
<tr>
<td>850</td>
<td>900</td>
<td>874.6</td>
<td>950</td>
</tr>
<tr>
<td>800</td>
<td>800</td>
<td>800</td>
<td>950</td>
</tr>
<tr>
<td>750</td>
<td>800</td>
<td>774.6</td>
<td>950</td>
</tr>
</tbody>
</table>
The settling time in this graph is about 25% of control signal period and frequency deviations are ±17.5 kHz and ±35 kHz. By decreasing cut-off frequency of LPF, we can decrease the frequency deviation of the output but the settling time of system will longer. These output parameters are in trade-off relationship with each other. The frequency deviation range can be calculated by using the result in Table 3.2.1 as explained in this equation.

\[
\text{Frequency deviation range} = \frac{(\text{deviation of VCO control signal} \times \text{sensitivity gain})}{2}
\]

The two deviation ranges shows that the PLL circuit synthesized the two different frequencies without frequency overlapping. These two ranges obviously separated each other.

Fig 3.2.8 (a) illustrates the power spectrum of synthesized frequencies (linear scale) in this model which can be done by keeping the data of the block “synthesized signal” in time domain and using Discrete Fourier Transform (DFT) theory to observe the output in frequency domain. The command file used to plot the spectrum is performed in Appendix A.1. The red color is the spectrum of 100MHz signal and blue color is the spectrum of 100.1MHz. The spectrum shown were normalized with the power of carrier signal which means that the carrier power equal to 1. Normalized power spectrum in log scale is also illustrated in Fig 3.2.8(b). One spurious signal occurred at 100.2 MHz in -18 dB level which this level was around the level at mid-band of two signals (at 100.05 MHz, -20 dB level). Signal to noise ratio (S/N) was about 20 dB. Noise in this ratio was calculated from the synthesis itself which not include noise from channel when transmit the signal. The level of acceptable noise sideband and spurious signal was not assigned in this step; it can be any value as much as the receiver can recover the carrier signal from received signal. S/N ratio and bit error rate of data transmitting was not discussed in this part because the transmitting receiving sides were not simulated simultaneously.
After the PLL model in Fig 3.2.7 had been satisfied the requirement, the synthesized frequencies were used as clock signal of data buffer (FIFO) to drive out data signal. The complete model for transmitting part in Fig 3.2.9 shows the data signal generator with 100Mbps was added into data buffer and was driven out by synthesized signal from PLL. The power spectrum of output signal from data buffer is shown in Fig 3.2.10, the spectrum shows that there is no spectrum, peak of clock frequency signal because the data signal is non-return to zero (NRZ) pattern. However the circuit in receiving section can recover the clock signal back from the transmitting signal.

![Power Spectrum of Synthesized Frequencies](a)

![Power Spectrum of Synthesized Frequencies](b)

Fig 3.2.8 Power spectrum of two different synthesized frequencies (a) linear scale (b) log scale
3.3 Phase Locked Loop Model in Receiving Section

In this part, PLL circuit was designed as clock recovery to retrieve the clock signal from received signal. This is because received signal is NRZ pattern, with no
accompany clock signal. The designed clock recovery model is shown in Fig 3.3.1. Because of receiving section was not simultaneously simulated with transmitting model, the transmitting signal was modeled as signal (NRZ(non return to zero) pattern) with frequency equal to 100MHz or 100.1MHz dependent to control signal. In this model, the initial frequency of VCO (frequency generates from VCO when 0 V is applied) was set at 100MHz with sensitivity gain 10^5 MHz/V. The simulation result in Fig 3.3.2 shows the two different voltage levels which mean two clock signals were recovered from received signal depends on control signal in transmitting section. However, the sensitivity gain used in this model is very small. A commercial VCO which has the oscillation range around 100 MHz almost all has the sensitivity gain around 8-10 MHz/V. About 40dB of gain has to be attenuated which is not practical due to the high power supply noise. Other type of phase detector was changed to the model to observe and find the optimal results as illustrated in Fig 3.3.3

![Clock recovery model by PLL](image)
Fig 3.3.2 Control signal in clock recovery 1st model

**Changed phase detector**

Fig 3.3.3 Clock recovery model with changing PD

Fig 3.3.4 shows the VCO control signal in PLL loop after change the phase detector which two levels of output voltage clearly separated. High fluctuation occurred at the beginning of each control bit but the signal can reach stable state within 25% of control data period. Locked-in range of output is about ±20 kHz. Power spectrum of two recovered signals which were normalized to 1 at carrier frequencies show their peaks are in the point of 100MHz and 100.1MHz in Fig 3.3.5(a) in linear scale and (b) in log scale. The recovered clock signals are the function of control signal; using some circuits can retrieve the control signal from clock signal.

However, these output results are just from the simulation in ideal case, no jitter or phase noise included. In actual circuit operation, there are noises from each component. Therefore, the results in actual circuit will be a bit poorer than that of simulation.
Fig 3.3.4 Output signal of LPF of model in Fig 3.12

(a)      (b)

Fig 3.3.5 Power spectrum of two different recovered signals
(a) linear scale   (b) log scale
Chapter 4
LSI Design Process

4.1 VHDL Coding and Simulation Results

The digital parts of PLL model were designed using LSI design process. At first, VHDL hardware language was used to design the behavioral level of digital parts which consist of frequency divider and phase frequency detector. For the frequency divider, divide number are 100 for control signal equal to ‘0’ and 100.1 for control signal equal to ‘1’. The 100 divider was created from counter by counting the edge of input signal up to 100 and changing state in every half of divide number (100). For 100.1 divider, the divide number is fractional therefore the divider was created indirectly by dividing the input signal by 100 for 9 clocks of reference signal and dividing by 101 for 1 clock of reference signal in every 10 clocks of reference signal as explained in chapter 3. Fig 4.1.1 shows flowchart of frequency divider function which starts from creating two counters, “count” and “count_carry”. The “count_carry” used to count the rising edge of signal “ref_signal” until its value equal to 9 in the case of control bit equal to “1” and will keep the “count” = “0” when control bit equal to “0”. When the “count” equal to 9, the signal “carry” will be “1” otherwise it will be “0”. The “count” signal used to count the rising edge of signal “clk_in” until 98 when signal “carry” equal to “0” and until 99 when “carry” equal to “1”. The complete VHDL code for frequency divider and phase frequency detector are shown in Appendix A2-A7.
Fig 4.1.1 Frequency divider design flowchart
VHDL code for divider was simulated by using Synopsys Scirocco software. Fig 4.1.2 shows the symbol of divider design which consists of four input signals and one output signal.

![Symbol of frequency divider](image)

**Fig 4.1.2 Symbol of frequency divider**

The signal “carry”, the internal signal as explained above is the signal used to control the dividing process. The simulation result in Fig 4.1.3 shows that, if the control signal equal to “0” the carry signal equal to “0”. If the control signal equals to “1” the carry signal equal to “0” for 9 clocks and “1” for 1 clock in every 10 clocks of reference signal.

![Simulation result of frequency divider No.1](image)

**Fig 4.1.3 simulation result of frequency divider No.1**

Fig 4.1.4 shows that, if carry equal to “0”, the internal signal “clkout1” will change state when the counter (signal “count”) equal to “99” (the counter start from “0”). The output signal “clkout” be taken out after the signal “clkout1” one clock of signal “clkin” to prevent the spurious signal. For carry signal equal to “1” the internal signal “count” count up to “100” and the signal “clkout1” change the state at signal “count” equal to “100” or the 101st clock of signal “clkin” as shown in Fig 4.1.5.
The phase frequency detector was created from two D-flip-flop connected as shown in Fig 4.6 with the symbol of input and output signal in the design. From Fig 4.1.6 if frequency of “ref_signal” higher than of “div_signal” consequently only the output of upper D-flip-flop (Qd1 ,NQd1) are activated and the simulation result is illustrated in Fig 4.1.7. In opposite circumstance if frequency of “div_signal” higher than that of “ref_signal”, only the output of lower D-flip-flop (Qd2, NQd2) are activated as shown in Fig 4.1.8. The symbol of top level design in Fig 4.1.9(a) has the two sub designs of divider and phase detector as illustrated in Fig 4.1.9(b)
Fig 4.1.6 Symbol of phase frequency detector

Fig 4.1.7 Simulation results of PFD ($f_{\text{ref_signal}} < f_{\text{div_signal}}$)

Fig 4.1.8 Simulation results of PFD ($f_{\text{ref_signal}} > f_{\text{div_signal}}$)
4.2 Design Synthesis Process

After the functionality of the design meet the requirements, the design was then synthesized with considering timing constraint. The VHDL code is mapped into hardware logic gate for specific technology library. The design was synthesized by using Synopsys Design Analyzer software. To constrain the design, we have to create the clock which is/are associated with the clock pin of the design which will be used as the reference for timing analysis. In this design there are two clock signals which are “clkin” and “ref_signal”. The clock references for “clkin” were set at period 10 ns and 1000 ns for “ref_signal”. The delay times for input signals were specified to the design for analyzing the required time for a certain path. The operating conditions were both set as “worst condition” and “best condition” to observe for maximum delays and minimum delays of the design. After the design had been set the constraint, the synthesis process then had been executed. The synthesis timing report shows that there are no timing violations in this design therefore the design ready for design layout process. The synthesis report is illustrated in Appendix A.8.

4.3 Design Layout Process

In layout design process, many physical data were required such as the technology files which contain design rules, net resistance and capacitance, unit measurement, GDSII files which contain physical layout information and VHDL gate-level netlist file which contains connectivity information of the design. The 0.35 um technology was used in this layout process. When the physical data had been prepared already the layout process was then performed by using Synopsys Astro software. The process was started from the design flow as explained in Fig 4.3.1. After finishing the
layout, the layout was checked with the design rules that are specified by IC technology specification and also verified for equivalence between schematic counterpart and layout. After design layout had been verified, the GDSII-out file was then extracted at the end of process. The design layout is illustrated in Fig 4.3.2.
4.4 Design Implementation to FPGA

The VHDL designed code was also programmed to FPGA chip and then was implemented to the circuit board to use before the fabricated chip will be received. In this research, the FPGA model EP1C12Q240C8, Cyclone, Altera was used. The input and output ports were assigned to the I/O pin as in Table 4.4.1. After that the designed code was loaded to FPGA chip via AS mode.

Table 4.4.1 pin assignment of the design to FPGA chip

<table>
<thead>
<tr>
<th>pin name</th>
<th>pin number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Din</td>
<td>139</td>
</tr>
<tr>
<td>rst</td>
<td>131</td>
</tr>
<tr>
<td>control</td>
<td>158</td>
</tr>
<tr>
<td>ref_signal</td>
<td>170</td>
</tr>
<tr>
<td>clkin</td>
<td>121</td>
</tr>
<tr>
<td>Qd1</td>
<td>187</td>
</tr>
<tr>
<td>Qd2</td>
<td>194</td>
</tr>
<tr>
<td>NQd1</td>
<td>183</td>
</tr>
<tr>
<td>NQd2</td>
<td>186</td>
</tr>
</tbody>
</table>
5.1 Circuit Implementation of Transmitting Section

As explained in chapter 4 that the digital parts, consist of frequency divider and phase frequency detector were programmed to the FPGA already. Next step, the analog parts which consist of VCO, crystal oscillator and low pass filter were implemented. For VCO, crystal oscillator, commercial products were used while the low pass filter was implemented by connecting the resistor, capacitor and active components as the values were calculated from the simulation models.

**Voltage Controlled Oscillator (VCO)**

**VCO features**

- Frequency oscillation range 95 -120 MHz
- Tuning range 0-5 V
- Power supply 12 V, maximum current 20 mA
- Tuning sensitivity 7-10 MHz/V
- Power output +10 dBm

Before implement VCO with other components, VCO was tested to measure the actual values of tuning range, oscillation range and sensitivity gain. Table 5.1.1 illustrates the data of oscillation frequencies versus the applied tuning voltages. From this table, the actual frequency oscillation range is from 85.19-125.13 MHz. The frequency at 0V of tuning voltage is quite different from the datasheet. Fig 5.1.1 shows the graph of relation between oscillation frequencies versus applied voltages with calculated sensitivity gain. The values of tuning sensitivity were calculated from data in Table 5.1.1. The average sensitivity gain is 7.98 MHz/V. However, the gain was calculated in each period of tuning voltage and can be explained that the sensitivity gain is not exactly linear through the tuning range. Tuning voltage from 0-2 V, calculated gain is around 9 MHz/V while the gain decrease to 7.5MHz/V at the range from 3-5V.
Table 5.1.1 Experiment values of oscillation frequency of VCO versus the tuning voltage

<table>
<thead>
<tr>
<th>Tuning voltage (V)</th>
<th>Oscillation frequency (MHz)</th>
<th>Tuning voltage (V)</th>
<th>Oscillation frequency (MHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>85.19</td>
<td>2.8</td>
<td>109.21</td>
</tr>
<tr>
<td>0.3</td>
<td>87.89</td>
<td>3</td>
<td>110.66</td>
</tr>
<tr>
<td>0.5</td>
<td>89.71</td>
<td>3.3</td>
<td>112.84</td>
</tr>
<tr>
<td>0.8</td>
<td>92.31</td>
<td>3.5</td>
<td>114.3</td>
</tr>
<tr>
<td>1</td>
<td>94.12</td>
<td>3.8</td>
<td>116.52</td>
</tr>
<tr>
<td>1.3</td>
<td>96.97</td>
<td>4</td>
<td>117.9</td>
</tr>
<tr>
<td>1.5</td>
<td>98.83</td>
<td>4.3</td>
<td>120.09</td>
</tr>
<tr>
<td>1.8</td>
<td>101.45</td>
<td>4.5</td>
<td>121.54</td>
</tr>
<tr>
<td>2</td>
<td>103.07</td>
<td>4.8</td>
<td>123.72</td>
</tr>
<tr>
<td>2.3</td>
<td>105.45</td>
<td>5</td>
<td>125.13</td>
</tr>
<tr>
<td>2.5</td>
<td>107</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig 5.1.1 Graph of experiment values of oscillation frequency of VCO versus tuning voltage

The output signal from VCO oscillated at no dc offset while FPGA input port detect the signal from 0 V to 3.3 V, so the VCO output signal has to be shifted up to oscillate at 1.65 V dc level by connecting RC components as shown in Fig 5.1.2.
Fig 5.1.2 Circuit diagram for shifting dc level

**Low pass filter**

**2\textsuperscript{nd} Active low pass filter**

\[
F(s) = \frac{1}{s^2 R^2 C_1 C_2 + 2sRC_2 + 1}
\]

\[
F(S) = \frac{1}{1 + sR_x C_x}
\]

**Difference Amplifier**

\[
V_o = (NQd_2 - NQd_1) \cdot \frac{R_2}{R_1}
\]
**Operation Amplifier**

Op-amp was used in PLL for the two sub-circuits which were low pass filter and difference amplifier. Below is its specification from the datasheet.

**Op Amp Features**

- 120 MHz Bandwidth, Gain = -1
- 1.7 nV/√Hz Input Voltage Noise
- 1.5 pA/√Hz Input Current Noise
- Operation: specified for ±5V to ±15V
- ±3V Output Swing into a 150 Ω Load

**PLL circuit Implementation**

After the analog parts were connected as explained above, the FPGA chip that was programmed as divider and phase detector then were installed to the circuit board. The output from VCO after shifted up was sent to the FPGA (at “clkin” pin). 1MHz oscillator was connected to “ref_signal” pin. “rst” pin was connected to ground and “Din” pin was connected to 3.3 V. Two pairs of output pins which are “NQd1” with “NQd2” and “Qd1” with “Qd2” can be used to connect to difference amplifier. If use “NQd1” and “NQd2” pins, the “NQd1” pin will be attached to the negative port of op-amp while “NQd2” pin will be injected to positive port. Oppositely if use “Qd1” and “Qd2”, the connect ports will swap with the first case.

![Fig 5.1.5 Circuit implementation of transmitting section](image)
The actual PLL circuit which all part name are labeled is illustrated in Fig 5.1.5. While circuit was operating, oscilloscope was used to observe the output signal as shown in Fig 5.1.7 (a) but the detected signal and measured frequencies were not accurate due to the efficiency of measurement. The differences between using control bit “1” and bit “0” cannot be observed. Therefore the digital counter was used instead to measure the output frequency. When control signal was switched to bit “0”, the output frequency has the range of 100.00035 - 100.00037 MHz (triggered rate at 0.1s) while output signal has the range of 100.10036 - 100.10038 MHz when control bit was switched to “1” as demonstrated in Fig 5.1.7 (b) and (c) respectively. From the results, it shows that the PLL circuit can synthesize two frequencies dependent to the control bit. However, control data, applied to the circuit was only the switching between 3.3V and ground. This is because the period of control data is 0.1 ms while the smallest trigger rate of digital counter is 1ms. So the different of two output frequencies cannot be observed when the switching between control bit “1” and bit “0” is performed and also cannot measure the output settling time. We lacked the device which is more accurate than oscilloscope to observe the PLL performance in time domain.

The output signals also were observed their spectrum by using spectrum analyzer. Fig 5.1.8 shows the spectrum of 100 MHz frequency synthesis in linear scale which was captured from analyzer span at 500 kHz. Noise sidebands occurred and were observed in the spectrum which the value and their positions are declared in the figure. Spectrum of 100.1 MHz is illustrated in Appendix A.9 which has higher noise sideband compare to 100 MHz spectrum. One method to reduce noise generated in the circuit is narrow the cutoff frequency of LPF which is also affect in settling time of the system. From the circuit model in Chapter 3, 800 rad/s cutoff frequency was selected which made the system offer the settling with 25% of control period. LPF in the circuit was changed to 700 rad/s in order to improve the output spectrum which caused the settling time increased to 35% of control period. Slower settling time will take the effect to the receiving part which will be observed whether the receiver circuit can recovered the signal within the control period or not.
Fig 5.1.7  Experiment for transmitting system operation
(a) Experiment set up
(b) Synthesized frequency when control bit = ‘1’
(c) Synthesized frequency when control bit = ‘0’

Fig 5.1.8 Spectrum of 100 MHz frequency synthesis (used cutoff value from simulation)

A shot term random frequency fluctuations of signal is commonly used for describing the term “Phase noise” [11]. There are two types of fluctuating phase terms. First is Phase noise which is most often expressed as a ratio of sideband power in a 1 Hz bandwidth at an offset f Hz away from the carrier to the signal power as shown in Fig
5.1.9. With using spectrum analyzer, it displays the power for a certain resolution bandwidth (RBW) so the power in a 1-Hz bandwidth has to be subtracted with $10 \log_{10}(RBW)$. The result finally has to be added with the correction factor of the device.

![Fig 5.1.9 Phase noise measuring](image)

$$S_c(f) \text{ in dB } = 10 \times \log_{10}[S_c(f)], \text{ dBc/Hz}$$

**Fig 5.1.10 Power spectrum of 100 MHz frequency synthesis**

<table>
<thead>
<tr>
<th>Offset Value</th>
<th>Frequency (MHz)</th>
<th>Phase Noise (dBc/Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 kHz</td>
<td>99.990</td>
<td>-45.39</td>
</tr>
<tr>
<td></td>
<td>100.010</td>
<td>-48.84</td>
</tr>
<tr>
<td>25 kHz</td>
<td>99.975</td>
<td>-69.49</td>
</tr>
<tr>
<td></td>
<td>100.025</td>
<td>-67.74</td>
</tr>
<tr>
<td>50 kHz</td>
<td>99.950</td>
<td>-63.45</td>
</tr>
<tr>
<td></td>
<td>100.050</td>
<td>-57.06</td>
</tr>
<tr>
<td>100 kHz</td>
<td>99.990</td>
<td>-86.71</td>
</tr>
<tr>
<td></td>
<td>100.100</td>
<td>-83.21</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Spurious signals</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.1 99.767 MHz/-79.50 dBc</td>
</tr>
<tr>
<td>No.2 99.960 MHz/-55.47 dBc</td>
</tr>
<tr>
<td>No.3 100.042 MHz/-55.11 dBc</td>
</tr>
<tr>
<td>No.4 100.236 MHz/-78.87 dBc</td>
</tr>
</tbody>
</table>
Second is called “spurious signal” which appeared as distinct components in spectral density. It is expressed as a ratio of power in a particular spurious sideband to the power in the main output signal expressed in dBc.

At this step, after the cutoff frequency of the LPF was decreased, the output spectrum was observed again. Fig 5.1.10 and Fig5.1.11 show the output spectrum as control bit = ‘0’ and control bit = ‘1’ respectively which were measured in log scale, span with 500 kHz and 10 kHz RBW. Phase noise was measured at 4 offset values as declared in the figure. Discrete spurious signals appeared in the spectrum as shown in the list. In this measurement, 10 kHz RBW the relative power between carrier and sideband noise was subtracted by 10log (10k) and added with correction factor of spectrum analyzer 2 dB. By comparing phase noise between 100MHz and 100.1 MHz output at each offset, at 10 kHz and 25 kHz offset, phase noise of both outputs are almost equal. At 50 kHz offset, even phase noise of 100 MHz spectrum is higher than that of 100.1MHz, but it cannot clarify that which output shows better performance because spurious signal occurred in 100 MHz output spectrum at this offset. Similarly at 100 kHz offset, spurious signal appeared in 100.1 MHz spectrum at 100 kHz offset so phase noise measure at that offset is higher than of 100 MHz spectrum. Four spurious signals in appeared in 100MHz spectrum while six spurious lines occurred in 100.1MHz spectrum. Although the spurious
signal of 100.1 MHz spectrum is greater than that of 100 MHz but its highest level is less than of 100 MHz spectrum. The highest level of spurious in 100 MHz output is -55.11 dBC and that of 100.1MHz is only -69.32 dBC.

5.2 Circuit Implementation of Receiving Section

In this section, PLL circuit was implemented as clock recovery. Commercial products of PD and VCO were used. The feature of PD is explained briefly in next part and that of VCO has already illustrated in the section 5.1. The PLL circuit was implemented and was observed the output signal. The measured frequencies were not reached the desired values. Characteristic of each component was investigated in order to fix the problem. One reason was discovered due to the output characteristic of PD which is ranging from -1000mV to 1000 mV. Because of the tuning voltage of VCO is 0-5V and VCO oscillate 100 MHz frequency at tuning voltage around 1.65 V. Shifting the output characteristic was done by adding dc voltage with its output signal. Non-inverting summing amplifier circuit was utilized to this step and its circuit is illustrated in Fig 5.2.1[12]. Fig 5.2.2 shows the whole PLL circuit which all component values are labeled. Low pass filter used in this part is 2nd order active filter as same as in transmitting part.

The input signal used is the synthesized signal from PLL in transmitting part without the data signal included. Because we want to detect the control signal which was transmit in the form of two synthesized frequencies before managing the system with data signal. The PLL circuit in transmitting part was operated in order to transmit the signal to the receiver circuit as illustrated in Fig 5.2.3 a) for control bit “1” and Fig 5.2.3b) for control bit “0”.

Phase Detector Features

- Frequency (RF1 & RF2) 10-200 MHz
- Power in: 7 dBm
- Impedance (Output Load): 500Ω
- Output Polarity (RF1/RF2 in-phase): neg.
- DC output: Typical 1000 mV

![Non-inverting summing amplifier circuit](image-url)
Digital counter was used again to measure the output signal in both case of control bit. 100 MHz and 100.1 MHz can be detected as set control bit to “0” and “1” respectively as illustrated in 5.2.3 (a),(b). Two output spectrums were observed and illustrated in Fig 5.2.4 and Fig 5.2.5. Phase noise of 100 MHz and 100.1 MHz spectrum are almost equal at 10 kHz offset. At 25 kHz offset, phase noise of 100.1 MHz spectrum is about 5.19 m higher than phase noise of 100 MHz. Because spurious signal occurred in 100 MHz spectrum at 50 kHz and occurred in 100.1 MHz spectrum at 100 kHz offset, we cannot used the phase noise values at these levels determine the output performance. However, the highest level of spurious signal of 100 MHz spectrum is 18.7 dBm higher than the highest level of spurious signal of 100.1 MHz spectrum. By comparing the spectrum of transmitting and receiving parts, spurious signals appeared as almost the same position with almost the same level for 100 MHz spectrum. In contrast, the level of spurious signals in 100.1 MHz spectrum is quite high compare with the level in transmitting part. Phase noise and spurious signal in the receiving part occurred from each component in the circuit itself. However, the main cause is from the transmitting part.
signal because the output from frequency synthesizer circuit has high phase noise with a number of spurious signals.

![Power spectrum of 100 MHz recovered signal](image)

**Fig 5.2.4 Power spectrum of 100 MHz recovered signal**

**Problems**

From the output spectrum of both transmitting and receiving parts, we faced with the short frequency stability problem which is mainly from phase noise and spurious signal. In transmitting circuit, phase noise level at 10 kHz offset is -47 dBc/Hz. The highest spurious levels are -51.11 dBc and -69.32 dBc for 100MHz and 100.1 MHz spectrum respectively. These values are quite high which will take the effect to signal to noise ratio therefore the circuit has to be modified and improved in order to reduce the phase noise and spurious signal.

**Sources of noise**

Sources of noise in PLL system consist of noise from the reference generator, noise from the voltage-controlled oscillator and noise in the PLL system blocks. Noise
generated from PLL system blocks consist of loop filters, amplifier, phase detector and frequency divider[13]. For the spurious signal, it can be divided in to reference spurs and non-reference spurs. Reference spurs, the spurious that occur at the integer multiple of reference frequency are mainly produced by phase frequency detector because the reference signal and its harmonics leak through the loop filter. Non-reference spurs, the spurious sideband appearing at the output not related to the reference frequency usually occur at the fractional N PLL [14]. They occur at the multiples of the fractional modulus. Suppose the effective division factor represent by

\[
N + \frac{X}{Y}
\]

Spurious signal appear at

\[
n \frac{f_r}{Y}
\]

where \(f_r\) is reference frequency

Phase noise measured at 4 offset values

@ 10 kHz offset
- 100.090 MHz \(\Rightarrow\) -48.38 dBc/Hz
- 100.110 MHz \(\Rightarrow\) -46.11 dBc/Hz

@ 25 kHz offset
- 100.075 MHz \(\Rightarrow\) -67.94dBc/Hz
- 100.125 MHz \(\Rightarrow\) -63.85dBc/Hz

@ 50 kHz offset
- 100.050 MHz \(\Rightarrow\) -75.40dBc/Hz
- 100.150 MHz \(\Rightarrow\) -77.25dBc/Hz

@ 100 kHz offset
- 100.000 MHz \(\Rightarrow\) -73.58dBc/Hz
- 100.200 MHz \(\Rightarrow\) -73.85dBc/Hz

Spurious signals
- No.1 99.900 MHz/-77.82 dBc
- No.2 99.940 MHz/-74.96 dBc
- No.3 99.999 MHz/-73.93 dBc
- No.4 100.200 MHz/-74.04 dBc
- No.5 100.265 MHz/-74.51 dBc
- No.6 100.302 MHz/-78.78 dBc

Fig 5.2.5 Power spectrum of 100.1 MHz recovered signal
In this experiment, the fractional divider 100.1 was produced; with 1 MHz reference frequency therefore the spurious appeared at multiples of 100 kHz offset away from carrier frequency. Additionally non-reference spurs results from power supply noise also.

To minimize noise in the circuit, the following methods are proposed and should be done in the next experiment.

- The cutoff-frequency of LPF will be decreased within the acceptable of settling time.
- The digital parts will be re-designed to solve the problems of non-reference spurious signal.
- The whole circuit will be implemented to the printed circuit with proper layout and good grounding technique.
- The DC batteries will be used instead of the electric power supply device to provide the best overall performance.
Chapter 6
Conclusion

Transmitting and receiving parts of optical wireless access has been designed by utilizing phased locked loop circuit. Phase locked loop was demonstrated as frequency synthesizer in transmitting section and as clock recovery in the receiving section. Conclusion for each process which has been done in this research is explained below.

- The system levels of transmitting and receiving section has been designed. PLL which was modeled as frequency synthesizer in transmitting part can synthesize the desired frequencies dependent to control signal. For receiving part, PLL which was modeled as clock recovery can recover two different frequencies from transmitting signal. The settling time of each section is about 25% of control signal period.

- The digital parts which are fractional frequency divider and phase frequency detector has been designed by using VHDL hardware language and synthesized with considering timing constraint. The physical layout process was then performed and was checked with the design rules and verified for equivalence between schematic counterpart and layout. After design layout had been verified, it was sent into fabrication process.

- The digital parts were also installed to the FPGA chip and implemented to the circuit board with the analog parts for transmitting side. The circuit had been operated and observed the output performance. Digital frequency counter was used to measure the synthesized frequencies. The measured frequencies clarified that the PLL circuit can generate two different frequencies dependent to the control signal. However, the output spectrums appeared the high phase noise and spurious signal.

- Clock recovery circuit has been implemented and operated with the circuit in transmitting side. Only the clock signal in transmitting part was injected to the recovery circuit. The two frequencies output signal has been detected and observed their spectrum. The output frequencies measured by digital counter clearly distinguished. High level phase noise and spurious signals occurred in the output spectrum.

- The ways to improve and solve the problems of phase noise and spurious signal are proposed.
Future Works

• Improvement of output performance in transmitting and receiving section
• FIFO will be designed and implemented in the transmitting section
• Data signal will be transmitted through FIFO by using PLL output clock signal
• Implement the circuit to retrieve the control signal from the recovered clock signal and re-generate the data signal from the received signal.
References

A.1 Command files for power spectrum plotting

% N is Fourier transform length
% Fs is sampling frequency
% out is the output of the PLL model
% linear scale
function [Pyy,f] = spectrum(N,Fs,out)
    Y = fft(out,N);
    Pyy = (Y.*conj(Y)) ./N;
    f = (Fs)*(0:N/2 -1)/N;
    plot(f,(Pyy(1:N/2))./max(Pyy));

% log scale
function [Pyy,f] = spectrum(N,Fs,out)
    Y = fft(out,N);
    Pyy = Y.*conj(Y) / N;
    P = 10*log10(Pyy./max(Pyy));
    f = (Fs)*(0:N/2 -1)/N;
    plot(f,P(1:N/2))

A.2 VHDL file for Top_Level design

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

-- Uncomment the following lines to use the declarations that are
-- provided for instantiating Xilinx primitive components.
--library UNISIM;
--use UNISIM.VComponents.all;

entity PFD_DIV is
    Port ( Din : in std_logic;
           clkin : in std_logic;
           ref_signal : in std_logic;
           rst : in std_logic;
           control : in std_logic;
           NQd1 : out std_logic;
           NQd2 : out std_logic;
           Qd1 : out std_logic;
           Qd2 : out std_logic;
           div_signal : out std_logic
           )
end entity PFD_DIV;
architecture Struct of PFD_DIV is
component div100_101
PORT (  
    clkin : in STD_LOGIC;
    control : in STD_LOGIC;
            ref_signal : in STD_LOGIC;
    rst : in STD_LOGIC;
    clkout : out STD_LOGIC
);  
end component;
component PFD
PORT (  
    div_signal : in std_logic;
    ref_signal : in std_logic;
    Din : in std_logic;
    Qd1 : out std_logic;
    Qd2 : out std_logic;
    NQd1 : out std_logic;
    NQd2 : out std_logic)
end component;
signal clkout : STD_LOGIC ;
beginn  
U_div100_101 : div100_101
    port map ( clkin => clkin,
            control => control,
            ref_signal => ref_signal,
            rst => rst,
            clkout => clkout
    );
U_PFD : PFD
    port map ( ref_signal => ref_signal,
                div_signal => clkout,
                Din => Din,
                Qd1 => Qd1,
                Qd2 => Qd2,
                NQd1 => NQd1,
                NQd2 => NQd2
    );
div_signal <= clkout;
end STRUCT;
A.3 VHDL Test bench file for Top-Level design

-- VHDL Test Bench Created from source file pfd_div.vhd -- 02:55:04 09/09/2004
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

ENTITY pfd_div_tb_PFD_DIV_vhd_tb IS
END pfd_div_tb_PFD_DIV_vhd_tb;

ARCHITECTURE behavior OF pfd_div_tb_PFD_DIV_vhd_tb IS

COMPONENT pfd_div
PORT(
    Din : IN std_logic;
    clkin : IN std_logic;
    ref_signal : IN std_logic;
    rst : IN std_logic;
    control : IN std_logic;
    NQd1 : OUT std_logic;
    NQd2 : OUT std_logic;
    Qd1 : OUT std_logic;
    Qd2 : OUT std_logic
);
END COMPONENT;

SIGNAL Din : std_logic := '1';
SIGNAL clkin : std_logic := '1';
SIGNAL ref_signal : std_logic:='1';
SIGNAL rst : std_logic := '0';
SIGNAL control : std_logic := '1';
SIGNAL NQd1 : std_logic;
SIGNAL NQd2 : std_logic;
SIGNAL Qd1 : std_logic;
SIGNAL Qd2 : std_logic;
constant period1 :time:= 1 us;
constant period2 :time:= 0.2 ms ;
constant period3 :time:= 10.01 ns ;
BEGIN

uut: pfd_div PORT MAP(
    Din => Din,
    clkin => clkin,
    ref_signal => ref_signal,
    rst => rst,
    control => control,

    NQd1 => NQd1,
    NQd2 => NQd2,
    Qd1 => Qd1,
    Qd2 => Qd2
);

END ARCHITECTURE behavior;
\[ 
\begin{align*}
\text{NQd1} & \Rightarrow \text{NQd1}, \\
\text{NQd2} & \Rightarrow \text{NQd2}, \\
\text{Qd1} & \Rightarrow \text{Qd1}, \\
\text{Qd2} & \Rightarrow \text{Qd2}
\end{align*}
\]

); 
\text{ref\_signal} \Leftarrow \text{not\_ref\_signal after period1/2}; 
\text{control} \Leftarrow \text{not\_control after period2/2}; 
\text{clkin} \Leftarrow \text{not\_clkin after period3/2};

-- *** Test Bench - User Defined Section ***
\text{tb : PROCESS}
\text{BEGIN}
\text{wait; -- will wait forever}
\text{END PROCESS;}
-- *** End Test Bench - User Defined Section ***

END;
A.4 VHDL files for frequency divider (divider.vhd)

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

ENTITY div100_101 IS
PORT (
  clkin      : in  STD_LOGIC;
  rst        : in  STD_LOGIC;
  control    : in  STD_LOGIC;
  ref_signal : in STD_LOGIC;
  clkout     : out STD_LOGIC
);
end div100_101;

architecture Behavioral of div100_101 is

SIGNAL Count   : STD_LOGIC_VECTOR (6 Downto 0) := "0000000";
SIGNAL Clkout1 : STD_LOGIC;
SIGNAL Count_carry : STD_LOGIC_VECTOR (3 Downto 0);
SIGNAL carry : STD_LOGIC := '0' ;
BEGIN
  process(ref_signal,control,rst)
  begin
    if  (ref_signal'event and ref_signal ='1')  then
      if (rst='1') then
        count_carry <="0000" ;
      else
        if ( control ='1' and count_carry = "1001") then
          count_carry <= "0000";
        elseif (control ='1' and count_carry < "1001" ) then
          count_carry <= count_carry +1;
        else
          count_carry<="0000" ;
        end if;
      end if;
    end if;
  end process;
  carry <= '0' when (count_carry < "1001") else '1';
  PROCESS(clkin)
  BEGIN
    IF (clkin'event AND clkin='1') THEN
      IF (rst='1') THEN
        Count <= "0000000";
      ELSE
        count_carry <= "0000" ;
      end if;
    end if;
  end process;
end Behavioral;
-- If carry is 0 then count until 99
IF ((carry='0') AND (Count > "1100010")) THEN
  Count <= "0000000";
-- If carry is 1 then count until 100
ELSIF ((carry='1') AND (Count > "1100011")) THEN
  Count <= "0000000";
ELSE
  Count <= Count+1;
END IF;
END IF;
END PROCESS;

-- If Count less than 50 clkout is 1 else 0
clkout1 <= '1' when (Count < "0110010") else '0';

PROCESS(clkin)
BEGIN
  IF (clkin'event AND clkin='1') THEN
    clkout <= clkout1;
  END IF;
END PROCESS;
end Behavioral;
A.5 VHDL test bench file for frequency divider

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;
ENTITY div100_101_tb_div101_100_vhd_tb IS
END div100_101_tb_div101_100_vhd_tb;
ARCHITECTURE behavior OF div100_101_tb_div101_100_vhd_tb IS
  COMPONENT div100_101
    PORT(
      clkin : IN std_logic;
      control : IN std_logic;
      ref_signal : IN std_logic;
      rst : IN std_logic;
      clkout : OUT std_logic
    );
  END COMPONENT;
  SIGNAL clkin :  std_logic  := '1';
  SIGNAL control :  std_logic :='1';
  SIGNAL ref_signal :  std_logic :='1';
  SIGNAL rst :  std_logic := '1';
  SIGNAL clkout :  std_logic;
  constant period1 :time:= 1 us;
  constant period2 :time:=0.2 ms ;
  constant period3 :time:= 9.90 ns ;
BEGIN
  uut: div100_101 PORT MAP(
    clkin => clkin,
    control => control,
    ref_signal => ref_signal,
    rst => rst,
    clkout => clkout
  );
  ref_signal <= not ref_signal after period1/2;
  control <= not control after period2/2;
  clkin <= not clkin after period3/2;
  -- *** Test Bench - User Defined Section ***
  tb : PROCESS
  BEGIN
    wait; -- will wait forever
  END PROCESS;
  -- *** End Test Bench - User Defined Section ***
END;
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

-- Uncomment the following lines to use the declarations that are
-- provided for instantiating Xilinx primitive components.
--library UNISIM;
--use UNISIM.VComponents.all;
entity PFD is
  Port ( div_signal : in std_logic;
         ref_signal : in std_logic;
         Din  : in std_logic;
         Qd1  : out std_logic;
         Qd2  : out std_logic;
         NQd1 : out std_logic;
         NQd2 : out std_logic);
end PFD;
architecture Behavioral of PFD is
  signal q1: std_logic;
  signal q2: std_logic;
  signal rs1 : std_logic;
  signal rs2 : std_logic;
begin
  process(ref_signal,Din,rs1)
  begin
    if (rs1 = '0') then
      q1<= '0';
    else
      if ( ref_signal ='1' and ref_signal'event) then
        q1<= Din;
      end if;
    end if;
  end process;
  process(div_signal,Din,rs2)
  begin
    if (rs2 = '0') then
      q2<='0';
    else
      if ( div_signal ='1' and div_signal'event) then
        q2 <= Din;
      end if;
    end if;
  end process;
end Behavioral;
end process;
rs1 <= q2 nand q1;
rs2 <= rs1;
Qd1 <= q1;
Qd2 <= q2;
NQd1 <= not q1;
NQd2 <= not q2;
end Behavioral;
A.7 VHDL Test bench file for Phase frequency detector

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

ENTITY pfd_tb_PFD_vhd_tb IS
END pfd_tb_PFD_vhd_tb;

ARCHITECTURE behavior OF pfd_tb_PFD_vhd_tb IS

COMPONENT pfd
PORT(
  div_signal : IN std_logic;
  ref_signal : IN std_logic;
  Din : IN std_logic;
  Qd1 : OUT std_logic;
  Qd2 : OUT std_logic;
  NQd1 : OUT std_logic;
  NQd2 : OUT std_logic;
  R1 : OUT std_logic;
  R2 : OUT std_logic
);
END COMPONENT;

SIGNAL div_signal :  std_logic :='1';
SIGNAL ref_signal :  std_logic:='1';
SIGNAL Din :  std_logic := '1';
SIGNAL Qd1 :  std_logic ;
SIGNAL Qd2 :  std_logic ;
SIGNAL NQd1 :  std_logic;
SIGNAL NQd2 :  std_logic;
SIGNAL R1 :  std_logic;
SIGNAL R2 :  std_logic;
constant period1 :time:=1 us;
constant period2 :time:=1.1 us;

BEGIN

uut: pfd PORT MAP(
  div_signal => div_signal,
  ref_signal => ref_signal,
  Din => Din,
  Qd1 => Qd1,
  Qd2 => Qd2,
  NQd1 => NQd1,
NQd2 => NQd2,
R1 => R1,
R2 => R2
);
ref_signal <= not ref_signal after (period1)/2;
div_signal <= not div_signal after (period2)/2;

-- *** Test Bench - User Defined Section ***
tb : PROCESS
BEGIN
  wait; -- will wait forever
END PROCESS;
-- *** End Test Bench - User Defined Section ***
A.8 Timing Report of design synthesis of digital part

Information: Updating design information... (UID-85)

******************************************************************************
Report : area
Design : PFD_DIV/U_div100_101 (div100_101)
Version: 2003.03
Date : Sat Jan 8 10:55:24 2005
******************************************************************************
Library(s) Used:
   ROHM035_L (File: /usr1/libraries/rohm035/kyoto/rohm035_l.db)
Number of ports:                5
Number of nets:                70
Number of cells:               60
Number of references:      16
Combinational area:          5133.240234
Noncombinational area:    3439.799805
Net Interconnect area:      undefined  (Wire load has zero net area)
Total cell area:          8573.04039
Total area:                 undefined
1
design_analyzer>
******************************************************************************
Report : transitive_fanout
   -clock_tree
Design : PFD_DIV/U_div100_101 (div100_101)
Version: 2003.03
Date : Sat Jan 8 10:55:24 2005
******************************************************************************
No sources.
1
design_analyzer>
******************************************************************************
Report : timing
   -path full
   -delay max
   -max_paths 1
Design : PFD_DIV
Version: 2003.03
Date : Sat Jan 8 10:55:24 2005
******************************************************************************
Operating Conditions: BEST   Library: ROHM035_L
Wire Load Model Mode: top
Startpoint: U_div100_101/Count_reg[0]
(rising edge-triggered flip-flop clocked by clkin)
Endpoint: U_div100_101/Count_reg[6]
(rising edge-triggered flip-flop clocked by clkin)
Path Group: clkin
Path Type: max

Des/Clust/Port Wire Load Model Library
-----------------------------------------------
PFD_DIV 1k ROHM035_L

Point Incr Path
-----------------------------------------------
clock clkin (rise edge) 0.00 0.00
clock network delay (ideal) 0.00 0.00
U_div100_101/Count_reg[0]/C (LDFP010) 0.00 0.00 r
U_div100_101/Count_reg[0]/Q (LDFP010) 0.54 0.54 f
U_div100_101/add_56/plus/plus/A[0] (div100_101_DW01_inc_7_0) 0.00 0.54 f
U_div100_101/add_56/plus/plus/U1_1_1/CO (LHAD1P010) 0.35 0.88 f
U_div100_101/add_56/plus/plus/U1_1_2/CO (LHAD1P010) 0.33 1.22 f
U_div100_101/add_56/plus/plus/U1_1_3/CO (LHAD1P010) 0.33 1.55 f
U_div100_101/add_56/plus/plus/U1_1_4/CO (LHAD1P010) 0.33 1.88 f
U_div100_101/add_56/plus/plus/U1_1_5/CO (LHAD1P010) 0.33 2.21 f
U_div100_101/add_56/plus/plus/U5/Y (LXOR2P010) 0.23 2.44 f
U_div100_101/add_56/plus/plus/SUM[6] (div100_101_DW01_inc_7_0) 0.00 2.44 f
U_div100_101/U115/Y (LAND2P010) 0.24 2.68 f
U_div100_101/Count_reg[6]/D (LDFP010) 0.00 2.68 f
data arrival time 2.68

clock clkin (rise edge) 10.00 10.00
clock network delay (ideal) 0.00 10.00
U_div100_101/Count_reg[6]/C (LDFP010) 0.00 10.00 r
library setup time -0.51 9.49
data required time 9.49

data required time 9.49
data arrival time -2.68

slack (MET) 6.81

Startpoint: U_div100_101/Count_carry_reg[0]
(rising edge-triggered flip-flop clocked by ref_signal)
Endpoint: U_div100_101/Count_carry_reg[0]
(rising edge-triggered flip-flop clocked by ref_signal)

Path Group: ref_signal
Path Type: max

<table>
<thead>
<tr>
<th>Des/Clust/Port</th>
<th>Wire Load Model</th>
<th>Library</th>
</tr>
</thead>
<tbody>
<tr>
<td>PFD_DIV</td>
<td>1k</td>
<td>ROHM035_L</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>Incr</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>clock ref_signal (rise edge)</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>clock network delay (ideal)</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>U_div100_101/Count_carry_reg[0]/C (LDFP010)</td>
<td>0.00</td>
<td>0.00 r</td>
</tr>
<tr>
<td>U_div100_101/Count_carry_reg[0]/Q (LDFP010)</td>
<td>0.69</td>
<td>0.69 r</td>
</tr>
<tr>
<td>U_div100_101/U103/Y (LNAND2P010)</td>
<td>0.09</td>
<td>0.78 f</td>
</tr>
<tr>
<td>U_div100_101/U101/Y (LOR3P010)</td>
<td>0.65</td>
<td>1.43 f</td>
</tr>
<tr>
<td>U_div100_101/U98/Y (LNAND3P010)</td>
<td>0.27</td>
<td>1.69 r</td>
</tr>
<tr>
<td>U_div100_101/U97/Y (LNOR2P020)</td>
<td>0.08</td>
<td>1.77 f</td>
</tr>
<tr>
<td>U_div100_101/U96/Y (LAND2P010)</td>
<td>0.27</td>
<td>2.05 f</td>
</tr>
<tr>
<td>U_div100_101/Count_carry_reg[0]/D (LDFP010)</td>
<td>0.00</td>
<td>2.05 f</td>
</tr>
<tr>
<td>data arrival time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>clock ref_signal (rise edge)</td>
<td>1000.00</td>
<td>1000.00</td>
</tr>
<tr>
<td>clock network delay (ideal)</td>
<td>0.00</td>
<td>1000.00</td>
</tr>
<tr>
<td>U_div100_101/Count_carry_reg[0]/C (LDFP010)</td>
<td>0.00</td>
<td>1000.00 r</td>
</tr>
<tr>
<td>library setup time</td>
<td>-0.51</td>
<td>999.49</td>
</tr>
<tr>
<td>data required time</td>
<td></td>
<td>999.49</td>
</tr>
</tbody>
</table>

slack (MET) 997.45

Fig A.9 Spectrum of 100.1 MHz frequency synthesis

Fig A.10 Voltage spectrum of 100 MHz frequency synthesis

Spectrum of VCO output (linear scale)
@ 25 kHz offset
  99.975 MHz → 8.7 mV
  100.025 MHz → 11.85 mV
@ 50 kHz offset
  99.950 MHz → 5 mV
  100.050 MHz → 3.7 mV
@ 100 kHz offset
  99.900 MHz → 0 V
  100.100 MHz → 0 V
Fig A.11 Voltage spectrum of 100.1 MHz frequency synthesis

<table>
<thead>
<tr>
<th>Frequency Offset</th>
<th>Signal Amplitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>25 kHz</td>
<td>54.60 mV</td>
</tr>
<tr>
<td>50 kHz</td>
<td>44.40 mV</td>
</tr>
<tr>
<td>100 kHz</td>
<td>26.65 mV</td>
</tr>
</tbody>
</table>

Spectrum of VCO output (linear scale)
@ 25 kHz offset
100.075 MHz → 54.60 mV
100.125 MHz → 44.40 mV

@ 50 kHz offset
100.050 MHz → 8.30 mV
100.150 MHz → 8.90 mV

@ 100 kHz offset
100.000 MHz → 26.65 mV
100.200 MHz → 26.65 mV

Fig A.12 Voltage spectrum of 100 MHz recovered signal

Spectrum of VCO output (linear scale)
@ 25 kHz offset
99.975 MHz → 8.50 mV
100.025 MHz → 10.73 mV

@ 50 kHz offset
99.950 MHz → 14.33 mV
100.050 MHz → 13.89 mV

@ 100 kHz offset
99.900 MHz → 0 V
100.100 MHz → 0 V

Fig A.13 Voltage spectrum of 100.1 MHz recovered signal

Spectrum of VCO output (linear scale)
@ 25 kHz offset
100.075 MHz → 31.19 mV
100.125 MHz → 36.68 mV

@ 50 kHz offset
100.050 MHz → 5.29 mV
100.150 MHz → 4.65 mV

@ 100 kHz offset
100.000 MHz → 16.36 mV
100.200 MHz → 14.98 mV