2012

Design a real-time system for measuring the alpha factor of semiconductor lasers

Yuan Sun
University of Wollongong

Recommended Citation
UNIVERSITY OF WOLLONGONG

COPYRIGHT WARNING

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site. You are reminded of the following:

Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material. Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.
Design a Real-Time System for Measuring the Alpha Factor of Semiconductor Lasers

A thesis submitted in fulfillment of the requirements for award of the degree

Master of Engineering by Research
from
UNIVERSITY OF WOLLONGONG
by
Yuan Sun

School of Electrical, Computer and Telecommunications Engineering

July 2012
Dedicated to my family
Declaration

This is to certify the work reported in this thesis was done by the author, unless specified otherwise, and the no part of it has been submitted in a thesis to any other university or similar institution.

Signature:____________________

Yuan Sun

24 July 2012
Abstract

Linewidth enhancement factor, also called $\alpha$, is a key parameter of a semiconductor laser (SL). This factor determines the characteristic of SL, such as the linewidth, the chirp, the injection lock range in an SL, the response of the SL to external optical feedback. Measurement of $\alpha$ has been attracting many researchers’ attention. Different approaches have been developed for obtaining accurate value of $\alpha$. Self-mixing Interferometry (SMI) technique is thought as a simple and reliable method used for $\alpha$ measurement. In general, an SMI based sensing system consists of an SL, a lens and an external target. The information of the parameters associated with the system such as $\alpha$, feedback level factor (denoted as $C$) as well as external target are carried in the laser power(called SMI signal) emitted by the SL. The information can be retrieved by applying advanced signal processing on SMI signals.

Regarding $\alpha$ measurement, the existing SMI-based algorithms for measurement of $\alpha$ all have some limitations in terms of measure accuracy, feedback level range and measurement speed. In this thesis, a wavelet-based filtering algorithm is first developed for enhancing the quality of an SMI signal so that the accuracy of $\alpha$ retrieved from the SMI can be improved. Secondly, an improved algorithm is proposed for $\alpha$ measurement under a wide feedback level range up to $C = 6$. Finally, a real-time
measurement system for $\alpha$ is implemented by employing FPGA-based technique. The system is able to provide $\alpha$ value and $C$ value with fast measurement speed and satisfied accuracy.
Acknowledgement

First and foremost, I would like to express my sincere gratitude and respect to my supervisors, Dr. Yanguang Yu and Professor Jiangtao Xi, for their encouragement and guidance. Without their instructive suggestions, patient instructions, insightful criticism and expert guidance, this thesis cannot reach its present form. Their academic attitude and enlightenment not only help me with this thesis but also in my future study and career.

Secondly, I feel grateful to all the staff in School of Electrical, Computer and Telecommunications Engineering (SECTE) and Information and Communication Technology Research (ICTR) Institute who have helped me to build up an academic ability. Besides, I owe much to the students from Optoelectronic Signal Processing Research (OSPR) Lab, for their valuable advice and great cooperation.

Last but not least, my special gratitude goes to my beloved parents for their loving support. Without their loving consideration and great confidence in me, I would have not been able to accomplish this work.

Y. Sun, Y. Yu and J. Xi, "Wavelet transform based de-noising method for self mixing interferometry signals", Third Asia Pacific Optical Sensors Conference, Sydney, Australia, January 2012.
Table of Contents

Chapter 1  Introduction ........................................................................................................ 1
  1.1 Introduction to \( \alpha \) measurement ........................................................................ 2
  1.2 Self-mixing interferometry based approaches ...................................................... 9
  1.3 Existing problems ................................................................................................. 24
  1.4 Aims of thesis ....................................................................................................... 25
  1.5 Contributions ........................................................................................................ 25
  1.6 Organization of thesis ......................................................................................... 26

Chapter 2  Pre-processing of SMI signals ....................................................................... 29
  2.1 Noise features of SMI signals .............................................................................. 30
  2.2 Filter design .......................................................................................................... 32
  2.3 Performance test .................................................................................................. 44
  2.4 Summary ................................................................................................................ 48

Chapter 3  Improved algorithm for \( \alpha \) measurement ................................................ 49
  3.1 Analysis of SMI signal waveform ....................................................................... 50
  3.2 Algorithm description .......................................................................................... 51
  3.3 Simulation Verification ......................................................................................... 57
  3.4 Experimental Verification .................................................................................... 60
3.5 Summary ........................................................................................................ 62

Chapter 4  Real-time system................................................................................... 65

4.1 Introduction to FPGA...................................................................................... 66

4.2 System description ....................................................................................... 68

4.3 System Implementation .............................................................................. 69

4.4 System test .................................................................................................. 96

4.5 Summary ...................................................................................................... 98

Chapter 5  Conclusion ......................................................................................... 99

5.1 Research contributions .............................................................................. 99

5.2 Suggested future work ............................................................................... 100

Appendix A: Reference ....................................................................................... 101
List of Figures

Figure 1-1: Experiment set-up in [18]..........................................................................................5

Figure 1-2: The upper and lower detuning ranges........................................................................5

Figure 1-3: Experimental set-up in [6]. APD and BS are avalanche photodiode and beam splitter;
respectively. PZT is piezoelectric transducer................................................................................7

Figure 1-4: Experiment set-up in [7]. SL: a laser diode, CL: a collimating lens, HWP: a half wave plate,
M: a full mirror, HG: a holographic grating, CD & TC: the current driver and temperature
controller........................................................................................................................................8

Figure 1-5: (a) Experimental setup for linewidth measurements using diffraction grating and lens
system for mode selection and confocal mirror FP interferometer; (b) experimental setup for
linewidth measurements using Michelson interferometer ................................................................9

Figure 1-6: Two-mirror Fabry-Perot cavity model..........................................................................11

Figure 1-7: Basic structure of an SMI system................................................................................16

Figure 1-8: Characteristic of SMI signals in different regimes: (a) vibration trace of target; (b) an SMI
signal in weak feedback regime; (c) an SMI signal in moderate feedback regime; (d) an SMI
signal in strong feedback regime................................................................................................18

Figure 2-1: Characteristics of an SMI signal under moderate feedback........................................30

Figure 2-2: An experimental SMI signal.........................................................................................31
Figure 2-3: Flow chart of the wavelet transform based filtering process.

Figure 2-4: (a) A simulation SMI signal with $C = 3$ and $\alpha = 3$; (b) simulation signal with noise at SNR of 20 dB.

Figure 2-5: Daubechies 4-tap wavelet base and scaling base.

Figure 2-6: (a) Low frequency content of the SMI signal; (b) high frequency content of the SMI signal at first level; (c) high frequency content of the SMI signal at second level.

Figure 2-7: (a) Derivative of the simulation SMI signal; (b) the spatial filter based on the information from the derivative of the SMI signal; (c) filtered high frequency content at first level scale.

Figure 2-8: Filtering results of a simulation noisy SMI signal: (a) a simulation pure SMI signal; (b) the SMI signal with noise (SNR = 20 dB); (c) filtered SMI signal; (d) an enlarged transitions parts before (left subfigure) and after (right subfigure) the filtering.

Figure 2-9: Enlarged sparkle-like noise of the SMI signal shown in Fig. 2-6.

Figure 2-10: Comparison of performances of median filters with different filter lengths: (a) experimental signal; (b) filtered signal when N=5; (c) enlarged transition of the signal in (b); (d) filtered signal when N=10; (e) enlarged transition of the signal in (d); (f) filtered signal when N=20; (g) enlarged transition of the signal in (f).

Figure 2-11: Experimental set-up of the SMI system.

Figure 2-12: Filtering results of an experimental SMI signal: (a) original experimental SMI signal; (b) filtered SMI signal; (c) enlarged transition before (left subfigure) and after (right subfigure) filtering.

Figure 3-1: Hysteresis phenomenon of an SMI signal, where $C = 2$ and $\alpha = 5$ [60].
Figure 3-2: An SMI signal with $C = 2$ and $\alpha = 5$, used to show time intervals $t_{CB}$, $T_1$, $t_{AD}$ and $T_2$.

Figure 3-3: Trend of change in SMI signal waveform: (a) an SMI signal with $C = 3$ and $\alpha = 5$; (b) an SMI signal with $C = 4$ and $\alpha = 5$; (c) an SMI signal with $C = 5$ and $\alpha = 5$; (d) an SMI signal with $C = 6$ and $\alpha = 5$.

Figure 3-4: Flow chart of the proposed method.

Figure 3-5: Waveforms of $g(n)$ and $\phi_\alpha(n)$ where $\alpha=5$ and $C=4$ : (a): $g(n)$; (b): $\phi_\alpha(n)$; (c): $\phi_\alpha(n)$; (d): $\phi_\alpha(n)$.

Figure 3-6: Simulation results of the proposed method.

Figure 3-7: Processed experimental signal.

Figure 3-8: (a) Recovered $\phi_\alpha$ and its spectrum; (b) $\phi_\alpha$ and its spectrum.

Figure 4-1: FPGA design flow.

Figure 4-2: Basic structure of the FPGA based SMI system.

Figure 4-3: Simulation results of the ADC module.

Figure 4-4: Basic structure of FPGA based SMI signal pre-processing.

Figure 4-5: Flow chart of the design of the median filter; $g[k]$ is a data sample in SMI signals.

Figure 4-6: Simulation results of the median filter module.

Figure 4-7: Design of FIR in four-tap Daubechies wavelet.

Figure 4-8: Simulation results of the FIR filter.

Figure 4-9: Simulation results of sharp transition detect module.

Figure 4-10: Design of down- and up-sampling using FIFO.
Figure 4-11: Simulation results of down- and up-sampling unit. ................................................................. 79

Figure 4-12: Simulation results of the normalization module. ........................................................................ 81

Figure 4-13: Illustration of the time intervals to be measured. ...................................................................... 82

Figure 4-14: State machine of FPGA design on measurement of $\alpha$. ........................................................ 83

Figure 4-15: Simulation results of the receive module. .................................................................................. 86

Figure 4-16: Simulation results of the initial module. .................................................................................... 87

Figure 4-17: Simulation results of window module. ....................................................................................... 89

Figure 4-18: Simulation results of the locating module. ................................................................................. 91

Figure 4-19: Simulation results of the measuring module. .............................................................................. 92

Figure 4-20: Simulation results of the CalphaLUT module. .......................................................................... 93

Figure 4-21: Simulation results of the measurement method. ....................................................................... 94

Figure 4-22: LCD character set. .................................................................................................................. 95

Figure 4-23: Simulation results of the LCD module. ..................................................................................... 96

Figure 4-24: Measurement system based on FPGA. ....................................................................................... 97

Figure 4-25: An SMI signal shown on the oscilloscope. ................................................................................ 97

Figure 4-26: Measurement results on the LCD. .......................................................................................... 98
List of Tables

Table 1-1: Physical meanings of symbols in Equations (1.15)-(1.17)................................. 13
Table 1-2: Physical meanings of symbols in Equations (1.25)-(1.28)................................. 16
Table 2-1: Meanings of symbols in Fig. 2-3........................................................................... 35
Table 2-2: Absolute maximum ratings ($T=25 \, ^\circ C$) of HL7851G [56]................................. 46
Table 2-3: Electrical characteristics ($T=25 \, ^\circ C$) of HL7851G [56]................................. 47
Table 3-1: Comparison of $C$ range between the proposed method and the method [11]........... 55
Table 3-2: Comparison of measurement results of proposed method with method in [11]......... 62
Table 4-1: Definitions of ports in the ADC module............................................................... 70
Table 4-2: Definitions of ports in the down_up module......................................................... 79
Table 4-3: Definitions of ports in the normalization unit module............................................ 81
Table 4-4: Function description of each state in FPGA design.............................................. 84
Table 4-5: Definitions of ports in the initial module.............................................................. 86
Table 4-6: Definitions of ports in the window module.......................................................... 88
Table 4-7: Definitions of ports in the locating module.......................................................... 90
Table 4-8: Definitions of ports in measuring module........................................................... 91
Table 4-9: Definitions of ports in the CalphaLUT module..................................................... 93
Table 4-10: Definitions of ports in LCD module................................................................. 95
Chapter 1  Introduction

The linewidth enhancement factor, also called $\alpha$, is a key parameter of a semiconductor laser (SL), which determines the linewidth, the chirp, the injection lock range in an SL, and the response of the SL to optical feedback [1]. $\alpha$ is defined as the ratio of the partial derivatives, with respect to the carrier density, of the real and imaginary parts of the refractive index $n = n_r + in_i$ [2]:

$$\alpha = \frac{\partial n_r / \partial N}{\partial n_i / \partial N} = -2k_0 \frac{\partial n_r / \partial g}{\partial g / \partial N}$$  \hspace{1cm} (1.1)

where $n_r$ and $n_i$ are the real and imaginary parts of $n$, respectively. $N$ is the carrier density, $k_0$ is the free-space wave vector, and $g$ is the gain per unit length.

Due to the importance of $\alpha$ to a SL, researchers have conducted extensive investigation on $\alpha$ and developed different measurement methods in the past three decades, such as AM/FM method [3, 4], optical injection approach [5-7], optical feedback method [8-11], linewidth measurement [12] and other methods [13].

Among the existing methods, using optical feedback technique, the one called optical
feedback self-mixing interferometry (SMI) is thought as the simplest way to obtain \( \alpha \) with reliable value [2]. This thesis makes more in-depth investigation on SMI based method, including its performance improvement and real-time hardware design.

This chapter gives an introduction to this thesis. The literature review on existing measurement methods of \( \alpha \) is given in Section 1.1. In Section 1.2, SMI technique is described and SMI based measurements for \( \alpha \) are reviewed. The problems on \( \alpha \) measurement are identified in Section 1.3. The aims and the contributions of this thesis are presented in Section 1.4 and 1.5, respectively. The over-all structure of the thesis is described in Section 1.6.

1.1 Introduction to \( \alpha \) measurement

In past three decades, many researchers have studied on \( \alpha \) measurement extensively. The methods used for measuring \( \alpha \) can be described as AM/FM method [3, 4], optical injection approach [5-7], optical feedback method [8-11] and linewidth measurement [12]. These methods are introduced in this section in details.

1.1.1 AM/FM method

In 1996, Hua [3] developed a method to obtain \( \alpha \) and nonlinear gain of a vertical-cavity surface-emitting laser (VCSEL), which was using RF-modulation technique. The single-mode gain-guided VCSEL (4-QW, IO-pm window size) used in this experiment operates at \( \sim 840 \) nm at 300 K. Hua analyzed the single-mode rate equations used for
small-signal with the inclusion of phenomenological nonlinear gain term to the first order and obtain an equation [3]:

\[
\left( \frac{2\beta}{m} \right)^2 = \alpha^2 (1 + \varepsilon I_S)^2 \left[ 1 + \frac{(G_x I_S)}{\Omega^2} \right]
\]

(1.2)

where \( \beta \) is the phase-modulation index, \( m \) is amplitude modulation index, \( \varepsilon \) is the nonlinear gain coefficient, \( G_x \) is the nonlinear gain, and \( I_S \) is the steady-state value for intensity, \( \Omega \) modulation frequency. Since \( \varepsilon I_S \) is small enough to ignore [3], \( \alpha (1 + \varepsilon I_S) \) can be seen as \( \alpha \). \( \alpha \) can then be acquired from the measured normalized phase-modulation index \( \frac{2\beta}{m} \) versus modulation frequency \( \Omega \), and is estimated as \( 2.7 \pm 0.5 \). Also it is found that the measured nonlinear gain of the VCSEL device behaves differently from that of edge-emitting lasers with a growing in pump level.

The main difficulty of AM/FM method is that signal attenuation at high frequencies, which is a common issue in this method.

### 1.1.2 Optical injection approach

In 1990, Hui et al. [15] developed a method to measure \( \alpha \) in distributed feedback (DFB) lasers based on the principle of external optical injection locking. By analyzing the well know Van der Pol equations [16] in the stable locked state, where the frequency of slave laser is locked to that of the master, the half locking bandwidth is determined as

\[
\Delta \omega_m = (1 + \alpha^2)^{1/2} \rho \quad [16], \text{where} \quad \rho \quad \text{is the normalized injection level. When the phase}\n\]

\( \phi = 2\pi \), the zero change in the output optical power happens, leading to a frequency detuning \( \Delta \omega = \rho \). \( \alpha \) can then be obtained from
\[
\alpha = \left[ \left( \frac{\Delta \omega_m}{\Delta \omega_b} \right)^2 - 1 \right]^{1/2}
\] (1.3)

where \( \Delta \omega_m \) and \( \Delta \omega_b \) can be measured precisely under stable locked injection. The upper and lower detuning limits are researched and given in this paper. DFB-BH laser diode (Fujitsu FLD 150) with a wavelength of 1554 nm was used in the experiment and the value of \( \alpha \) was estimated as \( 5.5 \pm 0.6 \). The merits of the method are that it is simple and accurate, and that it does not need knowledge of the absolute amount of the injection level. On the other hand, it is not effective for Fabry-Perot (FP) lasers.

In 2001, Liu et al. [17] gave an idea to measure \( \alpha \) based on the relation between the upper and lower bounds of the locked and unlocked regimes. The experiment set-up is shown in Fig. 1-1. In this method, an asymmetric locking bandwidth is very important which is as below [18]:

\[
\Delta \omega_{\text{min}} = -\frac{c}{2n_o L} \sqrt{\frac{S_i}{S}} \left( 1 + \alpha^2 \right) < \Delta \omega < \frac{c}{2n_o L} \sqrt{\frac{S_i}{S}} = \Delta \omega_{\text{max}}
\] (1.4)

where \( S_i \) is injected optical power, \( S \) is test laser optical power, \( n_o \) is refractive index, \( L \) is laser cavity length and \( c \) is speed of light. The experiment is carefully controlled so that Eq. (1.4) can be applied. From Eq. (1.4), Fig 1-2 is obtained, where the upper and lower detuning ranges are plotted for the conditions that the injection-locking occurs, and the value of \( \alpha \) can be estimated from the ratio of the two slopes for the boundaries of the locking ranges. The value of \( \alpha \) measured at the laser wavelength of 1550 nm is 3.0, which agrees the typical value of long wavelength strained quantum wells lasers. Speaking of the advantages of this method, it has the merits of
injection locking technique, such as increasing the laser modulation bandwidth, decreasing the chirp, and measuring many fundamental parameters of SLs. Furthermore, it does not need the injection level. On the other hand, the limitation of this method is that it would bring uncertainty for different laser systems.

Figure 1-1: Experiment set-up in [17].

Figure 1-2: The upper and lower detuning ranges.
1.1.3 Optical feedback method

In 1989, Shin et al. [19] proposed a measurement method of $\alpha$ by optical self-locking. This method uses the asymmetry of the frequency locking range of a semiconductor laser with the optical feedback from the external confocal Fabry-Perot (CFP) cavity. The low and high frequency $\Delta v_{-}$, $\Delta v_{+}$, locking limits can be given by the following equations:

$$\Delta v_{-} = F\sqrt{1 + \alpha^2 + \Delta v_c / 2}$$  \hspace{1cm} (1.5)  

$$\Delta v_{+} = F + \Delta v_c / 2$$  \hspace{1cm} (1.6)

where $F$ is proportional to the power of the feedback light and $\Delta v_c$ is the full width at half maximum of the interferometer transmission line. The values of $\Delta v_{+}$, $\Delta v_{-}$ and $\Delta v_c$ can be measured and they yield the value of $\alpha$. The experiment set-up is shown in Fig. 1-3. The measured value of $\alpha$ is $3.07 \pm 0.26$, which is consistent with those measured by other methods.
In 2000, Shin et al. [11] developed two simple methods, a current scanning method and a reflectivity scanning method, to measure $\alpha$ with an external cavity semiconductor laser (SL). The functional relationship related to $\alpha$ between the emission wavelength and effective reflectivity of the compound cavity of the SL is the key to both of the two methods. Both of these two methods do not require modifying the laser configuration of traditional external cavity. The current scanning method uses the frequency tuning curve as a function of injection current to estimate the value of $\alpha$. A simple model which yields a linear relationship between the injection current and the phase shift of the incident wave after one round trip in the external cavity is introduced in the first method.

On the other hand, in the reflectivity scanning method, the external feedback intensity is scanned to control the effective reflectivity of SL which lead to the determination of the value of $\alpha$. In this paper, the modification of the external feedback intensity is implemented by adjusting the rotation angle of a half wave plate inserted in the external cavity. The experiment in this paper is shown in Fig. 1-4.
1.1.4 Linewidth measurement

In 1992, Toffano et al. [12] proposed a method to measure \( \alpha \) by linewidth measurement. The laser linewidth below and above threshold can be written as Eq. (1.7), (1.8), respectively.

\[
\Delta V_b = \frac{R_{th}}{2\pi P} \quad (1.7)
\]

\[
\Delta V_b = \frac{R_{th}}{4\pi P} (1 + \alpha^2) \quad (1.8)
\]

where \( R_{th} \) is the total spontaneous emission rate above threshold, \( P \) is the photon number in the mode. Comparing Eq. (1.7) and (1.8), \( \alpha \) is obtained as

\[
\alpha = \sqrt{\frac{2 \Delta V_u}{\Delta V_b} - 1} \quad (1.9)
\]

Two different experimental set-ups are undertaken, and they are shown in Fig. 1-5.

For single mode, the value of \( \alpha \) obtained by correlating Schawlow-Townes slopes above and below threshold is 2.6. For multimode, additional noise sources, such as partition noise, can be characterized introducing an additional factor.
Figure 1-5: (a) Experimental setup for linewidth measurements using diffraction grating and lens system for mode selection and confocal mirror FP interferometer; (b) experimental setup for linewidth measurements using Michelson interferometer

1.2 Self-mixing interferometry based approaches

SMI technique for sensing makes use of self-mixing effect. The SMI effect happens when a piece of laser light reflected or backscattered from a distant target ahead of the laser, re-enters the laser cavity, resulting in the variance of both the amplitude and the frequency of the lasing field. The modulated laser power, called SMI signal, is detected by a photodiode (PD). The first application of this effect represented in 1968, which was based on He-Ne, was developed to measure a shift of a moving target [20]. Since then, SMI based applications have been studied widely. SMI based technique can be used for measurements related to optical path displacement [21, 22], vibration [23-25], velocity [26-28] and physical parameters like linewidth enhancement factor $\alpha$ [8-10]. These
applications are reviewed in [29]. The applications have been extended to modal analysis, defect detection [30] and 3D profile [31]. The merits of the SMI sensing system are summarized in [32]:

- Optical part is minimal because it does not need external optical interferometer for the source;
- Set-up is self-aligned since the laser cavity itself acts like a filter and it can operate on a normal diffusive target surface;
- Information can be picked up both on the beam and the remote target;
- Sensitivity is very high up to sub-nanometer.

One of the mentioned applications, measurement of $\alpha$ is focused on in this thesis. In this section, a background of SMI technique is given. Firstly, the mathematical model of SMI effect is described. Then, basic structure of SMI system is depicted and the theory of SMI signals is introduced. The following part is the literature review on SMI based $\alpha$ measurement method. Since most SMI methods are based on the information retrieved from SMI signals, high-quality signals are required. Therefore, pre-processing of SMI signals is a necessary step before measurement. Literature review on pre-processing is given in the last part of this section.

### 1.2.1 Mathematical model

The mathematical model used to describe SMI sensing can be derived from two different ways. One way is to use three-mirror cavity to model the system. Another is using
1.2.1.1 Three mirror model

A single-mode laser conductor can be seen as a two-facet Fabry–Perot (FP) cavity and its graphic illumination is shown as Fig. 1-6 [33]:

![Figure 1-6: Two mirror Fabry-Perot cavity model.](image)

In Fig. 1-6, $f_1$ and $f_2$ are the two facets of the SL and $f_3$ is the surface of external cavity. $f_1$ and $f_2$ form the laser cavity, also called internal cavity. $f_2$ and $f_3$ form the external cavity. $c_1$, $c_2$ and $c_3$ stand for the amplitude reflection coefficient of $f_1$, $f_2$ and $f_3$. The internal cavity length and the external cavity length are represented by $d$ and $L$, respectively. In a general way, $|c_2| \ll |c_1|$. Thus the multiple reflection effect within the external cavity can be ignored. The electric field $E(t)$ undergoing a roundtrip within the compound cavity $f_1 \rightarrow f_2 \rightarrow f_3$ is described by Eq. (1.10) [34]:

$$E(t) = c_1 c_2 \exp \left\{ -j 4 \pi v \frac{nd}{c} + (g - \gamma) d \right\} E_0(t)$$

$$+ c_1 (1 - R_2) c_3 \exp \left\{ -j 4 \pi v \frac{nd + L}{c} + (g - \gamma) d \right\} E_0(t)$$

(1.10)

where $v$ is the optical frequency, $c$ is the speed of light in a vacuum, $R_2$ is $|c_2|^2$, $g$ is the linear gain per unit length due to the stimulated emission inside the laser cavity in
the presence of feedback, and $\gamma$ is any optical loss per unit length within the cavity, $E_0(t)$ is the initial electric field.

For a stationary stable laser oscillation, the amount of light amplified by the stimulated emission becomes equal to the total losses in the lasing system, which brings out the condition below [34]:

$$c_1[c_2 + (1 - R_2)r_2 \exp\{-j4\pi v \frac{L}{c}\}\cdot \exp\{-j4\pi v \frac{nd}{c}\} + (g - \gamma)d] = 1 \quad (1.11)$$

By solving this basic equation, the excess required gain $\Delta g$ and the additional phase $\phi_a(v)$ for the laser system in the presence of the feedback may be represented respectively as [34]:

$$\Delta g = g - g_0 = -\xi d \cos \phi_{\text{ext}} \quad (1.12)$$

$$\phi_a(v) = \xi \sin \phi_{\text{ext}} \quad (1.13)$$

where $g_0$ accounts for the linear gain in the absence of the feedback, $\xi$ is the coupling effect from the external reflection back into the laser cavity, and $\phi_{\text{ext}}$ denotes the phase of external reflection

$$\phi_{\text{ext}} = 4\pi v(L / c) \quad (1.14)$$

Equations (1.12) and (1.13) are usually employed to describe the basic characteristics of a diode laser in the presence of an external optical feedback.

### 1.2.1.2 Lang and Kobayashi model

The L-K equations which describe the dynamics of an SMI with feedback [22]. The equations are represented as below:
\[
\frac{d}{dt} E_0(t) = \frac{1}{2} \left[ G_N (N(t) - N_0) - 1/\tau_p \right] E_0(t) + \frac{\kappa}{\tau_L} E_0(t - \tau) \times \cos[\omega_o \tau + \phi(t) - \phi(t - \tau)] \]
\]

(1.15)

\[
\frac{d}{dt} \phi(t) = \frac{1}{2} \alpha G_N [N(t) - N_R] - \frac{\kappa}{\tau_L} E_0(t - \tau) \times \sin[\omega_o \tau + \phi(t) - \phi(t - \tau)] \]
\]

(1.16)

\[
\frac{d}{dt} N(t) = R_p - \frac{N(t)}{\tau_S} - G_N [N(t) - N_0] E_0^2(t)
\]

(1.17)

Table 1-1: Physical meanings of symbols in Equations (1.15)-(1.17).

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>(E_0(t))</td>
<td>Laser electric field.</td>
</tr>
<tr>
<td>(G_N)</td>
<td>Modal gain coefficient.</td>
</tr>
<tr>
<td>(N(t))</td>
<td>Average carrier (electron-hole pairs) density in the active layer.</td>
</tr>
<tr>
<td>(N_0)</td>
<td>Carrier density at transparency.</td>
</tr>
<tr>
<td>(\tau_p)</td>
<td>Photon lifetime, (1/\tau_p = G_N (N_R - N_0)).</td>
</tr>
<tr>
<td>(\kappa)</td>
<td>Feedback parameter.</td>
</tr>
<tr>
<td>(\tau_L)</td>
<td>Diode cavity round trip time.</td>
</tr>
<tr>
<td>(\tau)</td>
<td>External cavity round trip time.</td>
</tr>
<tr>
<td>(\omega_o)</td>
<td>Angular frequency of the unperturbed laser.</td>
</tr>
<tr>
<td>(\phi(t))</td>
<td>round trip phase</td>
</tr>
<tr>
<td>(\alpha)</td>
<td>(\alpha = (\partial \chi_e / \partial N) / (\partial \chi_l / \partial N)) with (\chi = \chi_e - i \chi_l), complex susceptibility.</td>
</tr>
<tr>
<td>(N_R)</td>
<td>Carrier density at threshold for the unperturbed laser.</td>
</tr>
<tr>
<td>(R_p)</td>
<td>Electric pumping term.</td>
</tr>
<tr>
<td>(\tau_S)</td>
<td>Carrier lifetime.</td>
</tr>
</tbody>
</table>

As to give a stationary solutions to Equations (1.15)-(1.17), \(E_0(t)\) and \(N(t)\) are set as constants which are \(E_C\) and \(N_C\), respectively. Furthermore, \(\phi(t)\) can be written as
\( \phi(t) = (\omega_F - \omega_b)t \) as the instantaneous optical frequency is \( \omega(t) = \omega_0 + [d\phi(t)/dt] \).

\( \omega_F = \omega_F(\tau) \) is the angular frequency of the laser with external feedback [22]. By substituting \( E_C \), \( N_C \), \( \phi(t) \), \( \omega_F \) and \( 1/\tau_p \) into Eq. (1.15), Eq. (1.18) can be obtained:

\[
N_C = N_T - \frac{2\kappa}{G_N\tau_L} \cos \omega_F \tau \tag{1.18}
\]

It shows that there is a modulation in the cavity density when the feedback exists.

Substitute \( E_C \), \( \phi(t) \), \( 1/\tau_p \) and Eq. (1.17) into Eq. (1.15), another relationship can be acquired:

\[
\omega_0 = \omega_F + \frac{\kappa}{\tau_L} (\alpha \cos \omega_F \tau + \sin \omega_F \tau) \tag{1.19}
\]

By introducing the feedback factor \( C \) [35, 36], Eq. (1.19) can be written as:

\[
\omega_0 = \omega_F \tau + C \sin(\omega_F \tau + \arctan \alpha) \tag{1.20}
\]

where \( C = \frac{\kappa \tau \sqrt{1 + \alpha^2}}{\tau_L} \). When \( C < 1 \), there is one solution for \( \omega_F \) in Eq. (1.20). And there are multiple solutions while \( C > 1 \). \( C=1 \) is the boundary between the weak and moderate feedback.

Then the electric field \( E_c^2 \) can be obtained from Eq. (1.16) with Eq. (1.17).

\[
E_c^2 = \frac{R_p \tau_s - N_T + \frac{2\kappa}{G_N\tau_L} \cos \omega_F \tau}{1 - \frac{2\kappa \tau_p}{\tau_L} \cos \omega_F \tau} \left( \frac{\tau_p}{\tau_s} \right) \tag{1.21}
\]

In practice, \( \kappa < 0.01 \). Thus, Eq. (1.21) can be approximated as

\[
E_c^2 = \frac{\tau_p}{\tau_s} \left( R_p \tau_s - N_T + \frac{2\kappa}{G_N\tau_L} \cos \omega_F \tau \right) \times \left( 1 + \frac{2\kappa \tau_p}{\tau_L} \cos \omega_F \tau \right) \tag{1.22}
\]
Let $E_{NF}^2 = \tau_p [R_p - (N_T / \tau_S)]$ stands for the stationary electric field in case of no feedback and second-order contributions be neglected, the output power $\Delta P$ due to feedback with respect to the unperturbed laser can be got [22],:

$$\Delta P \propto E_F^2 - E_{NF}^2 = \tau_p (R_p - N_0 / \tau_S) \frac{2\kappa \tau_p}{\tau_L} \cos \omega_F \tau$$  (1.23)

When assuming $\kappa$ does not depend on the external cavity, $\Delta P$ can be expressed as below:

$$\Delta P = \Delta P_{\text{max}} \cos \omega_F \tau$$  (1.24)

where $\Delta P_{\text{max}}$ is the maximum output power.

Equations (1.20) and (1.24) are used as the mathematical model for describe SMI sensing system.

### 1.2.2 Physical set-up and SMI signals

The basic structure of an SMI sensing system is shown in Fig. 1-7. The core part consists of an SL, a lens and an external target. When the target is moving, the output power $\Delta P$ from the SL will be modulated. This modulated power, called SMI signal, is detected by a PD, which is usually packed in the rear of an SL. Then $\Delta P$ is digitalized and processed by a personal computer (PC).

For the convenience of computer analysis on SMI signals, the mathematical model of SMI described by Equations (1.20) and (1.24) is rewritten in discrete form as follows. The explanation of each symbol is given in Table 1-2.
Figure 1-7: Basic structure of an SMI system

\[ \phi_f[n] = \phi_0[n] - C \sin(\phi_f[n]) + \arctan(\alpha) \]  \hspace{1cm} (1.25)

\[ P[n] = P_0(1 + m \times g[n]) \]  \hspace{1cm} (1.26)

\[ g(n) = \cos(\phi_f[n]) \]  \hspace{1cm} (1.27)

\[ \phi_0(n) = 4\pi L[n]/\lambda_0 \]  \hspace{1cm} (1.28)

Table 1-2: Physical meanings of symbols in Equations (1.25)-(1.28).

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Physical meanings</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n )</td>
<td>Discrete time index.</td>
</tr>
<tr>
<td>( \phi_f )</td>
<td>External light phases of a SL with feedback.</td>
</tr>
<tr>
<td>( \phi_0 )</td>
<td>External light phases of a free running SL.</td>
</tr>
<tr>
<td>( C )</td>
<td>Feedback level factor.</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>Linewidth enhance factor.</td>
</tr>
<tr>
<td>( P(n) )</td>
<td>Laser intensities with feedback.</td>
</tr>
<tr>
<td>( P_0 )</td>
<td>Laser intensities without feedback.</td>
</tr>
<tr>
<td>( m )</td>
<td>Modulation index.</td>
</tr>
<tr>
<td>( g(n) )</td>
<td>An SMI signal.</td>
</tr>
<tr>
<td>( L[n] )</td>
<td>External cavity length.</td>
</tr>
<tr>
<td>( \lambda_0 )</td>
<td>Emitted laser wavelength without feedback.</td>
</tr>
</tbody>
</table>

In SMI systems, two parameters are especially important, which are \( \alpha \) and \( C \). \( C \)
indicates the feedback level of the running SLs and characterizes the fringe shape of SMI signals. The behavior of SLs can be classified into different feedback regimes by different values of $C$ [37]. When $C$ is in the range of (0, 1), the SLs operate in weak feedback regime and the waveform of SMI signals is asymmetric sinusoidal like. When $1 < C < 4.6$, the SLs operate in moderate feedback regime and the waveform of SMI signals starts to exhibit hysteresis. Large $C > 4.6$ is referred as strong feedback regime. Figure 1-8 shows the characteristics of SMI signals in different feedback regimes, respectively.

Assuming the vibration of target is set as in Fig. 1-8(a), the expression of $\phi_0$ is $\phi_0 = A_0 + A \sin(2\pi f_s / f_s)$, where $A_0 = 4\pi L / \lambda_0$ and $A = 4\pi \Delta L / \lambda_0$, $\lambda_0$ is the emitted laser wavelength without feedback, $f_s$ and $f_s$ are vibration frequency and sampling frequency, respectively. Then SMI signals can be plotted using Equations (1.26)-(1.28).

From Fig. 1-8, it can be seen the SMI signal in weak feedback regime is like sinusoidal signals and in moderate and strong feedback regime, sharp transitions appear.
Figure 1-8: Characteristic of SMI signals in different regimes: (a) vibration trace of target; (b) an SMI signal in weak feedback regime; (c) an SMI signal in moderate feedback regime; (d) an SMI signal in strong feedback regime.

1.2.3 Literature review on SMI approaches

In 2004, Yu et al. [9] proposed a fast and easy method for the measurement of $\alpha$ based on self-mixing interferometry effect. In this paper, the authors firstly analyzed the solutions for the Lang-Kobayashi equations, and found out the relationship between the values of the parameters, $\alpha$ and $C$, and the time intervals between different characteristic points. The relationships can be represented as

$$\phi_{13} = \sqrt{C^2 - 1} + \frac{C}{\sqrt{1 + \alpha^2}} + \arccos(-\frac{1}{C}) - \arctan(\alpha) + \frac{\pi}{2}$$  \hspace{1cm} (1.29)

$$\phi_{24} = \sqrt{C^2 - 1} - \frac{C}{\sqrt{1 + \alpha^2}} + \arccos(-\frac{1}{C}) + \arctan(\alpha) - \frac{\pi}{2}$$  \hspace{1cm} (1.30)

Thus, by measuring the $\phi_{13}$ and $\phi_{24}$, $\alpha$ and $C$ can be calculated using these two
equations. This method can be applied to any single-mode SLs and the accuracy is estimated as $\pm 6.5\%$ after experiments. The most obvious advantage of this method is that it does not require the feedback strength which is difficult to obtain. But the limitation of this method is that it can only work in the moderate feedback regime when $1 < C < 3.6$.

In 2005, Xi et al. [8] developed a gradient-based optimization algorithm to measure $C$ and $\alpha$. The core algorithm of this paper is data fitting technique, which is to find $C$ and $\alpha$ so that the Equations (1.29) and (1.30) can match the data samples of the observed SMI signals. A cost function is defined to achieve the best match which is the summation of square errors between the observed data samples and the calculated ones using the model. The idea of this gradient-based technique is to update the two parameters towards their direction so that the cost function can be declined. The key to this process of the method is to update $C$ and $\alpha$ until it gives the satisfactory result. The advantages of this method are 1) It is very simple to implement; 2) This method can be used in all single-mode SLs running in weak feedback regime; 3) It is accurate because it employs all the data samples in the SMI signals. On the other hand, the main limitation of this method is that a pure harmonic vibration of the external target is necessary and another limitation is that this method only works in the weak feedback regime.

In 2005, Yu et al. [38] also proposed another method to measure multiple parameters including $\alpha$ using SMI technique. This method firstly obtains discrete SMI signals from observed output power of the laser. Then the vibration frequency and the phase
parameter are gotten using auto-correlation and phase unwrapping methods. Finally, the
data fitting techniques are employed to yield the values of the parameters and the
vibration information of a target including the frequency and the amplitude. The
disadvantage of this method is that it only covers the weak feedback regime.
In 2007, Yu et al. [10] developed an automatic measuring method of $\alpha$ using SMI
technique. This method is also based on data fitting technique as [8] and has the same
advantages as the method in [8]. Another important benefit is that this method does not
require the exact movement track of the external target, which reduces the work and
cost for the maintenance and calibration of the system. It considers all the parameters
from the L-K equations to estimate $\alpha$. The pre-steps of the method are to estimate the
vibration frequency and segment the SMI signal to obtain the initial phase. The
subsequent steps are summarized as following: 1) Set $C$ and $\alpha$ to minimize the cost
function and yield suitable $A$ and $A_0$; 2) Use $A$ and $A_0$ to minimize the cost
function and give reasonable $C$ and $\alpha$. These two steps can be repeated until the
parameters are stable. After different experiments, the standard deviation for estimated
$\alpha$ is less than 4.58% on average. It is a reliable measurement method, but the range of
this method is limited in the weak feedback regime.
In 2009, Wei et al. [39] presented a data-to-model fitting approach to estimate $\alpha$ using
SMI effect. The merits of this method are that it solves the limitations related to
data-to-model fitting approaches in [8, 10], the knowledge regarding target moving
trajectory and displacement. This method firstly segments the SMI signal as in [10] and
uses the same cost function with the methods in [8, 10]. Then investigation on the graphical study on the shape of the cost surface is applied and a genetic algorithm (GA) is employed since it is capable to achieve minimization with respect to four variables simultaneously. GA is the core part of this method and steps of it are summarized as: 1) generate initial population of 36 chromosomes randomly; 2) select the best fitted 18 chromosomes to the next generation and reproduce; 3) maintain the population of 36 by generating 18 off-springs by crossover and mutation to replace the discarded chromosomes; 4) GA keeps running until the terminating condition is met. By simulation and experiment test, this algorithm can achieve an accuracy of 3.8% for $\alpha$ measurement. However, this method still only works in weak feedback regime as the methods in [8, 10].

1.2.4 Literature review on pre-processing method

- **De-noising method of SMI signals combined by a median filter and a band-pass filter**

  In 2007, Yu et al. [40] proposed a de-noising method combined by a median filter and a band-pass filter with Kaiser window function. In this paper, the authors analyzed the characteristics of the noise which contaminate SMI signals and classified them into three types, high frequency noise, sparkle-like noise and slow-time fluctuation. The authors also pointed out the types of characteristic points including important information related to the measurement of parameters of SLs [6, 9] and the moving information of
the external target [8, 10]. Based on this analysis, Yu firstly uses a median filter to eliminate the sparkle-like noise and its length is fixed to 19-points. Then, a band-pass filter based on Kaiser window is employed to reduce the high-frequency noise. At last, these two filters are combined as a de-noising method. The authors tested their performance for measurement of $\alpha$, and this combining method is proved to be better than the two single filters. However, when the band-pass filter is adopted, the sharp transitions in the SMI signal waveform are smoothed and these changes of sharp transitions can decrease the accuracy of measurement of $C$ and $\alpha$ [9].

- **Pre-processing method for SMI signals using neural networks**

In 2007, Wei *et al.* [41] developed a neural signal interpolation technique for pre-processing for SMI signals. The aim of this method focuses on the accuracy of displacement and moving track information. Wei employs Radial Basis Functions (RBF) as the activation function for the neural network, which is efficient for interpolation and smoothing of data [42]. A RBF network consists of a hidden layer neurons linear outputs which is a sum of the weighted output from hidden layer. Root-mean-square error (RMSE) is employed for evaluating the quality of the network. When training the network, the network weights and parameters are updated so that RMSE can be minimized. After using this pre-processing method, the self-mixing signal is used to reconstruct the displacement with phase unwrapping method. The results are that the accuracy is $\lambda/25$ in the weak feedback regime and $\lambda/20$ in the moderate feedback regime, which were reliable. However, this method does not cover the strong feedback regime.
and it also can change the positions of sharp transitions which may induce the error for the measurement of $\alpha$ and $C$.

- **Multi-resolution signal decomposition: wavelet representation**

In 1989, Mallat [43] proposed a theory for multi-resolution signal decomposition, the wavelet representation. In this paper, the mathematical properties of the operator which transforms a function into an approximation at a resolution $2^j$ (j is an integer) are studied. Then the difference of information between two approximations at the resolutions $2^j$ and $2^{j+1}$ is extracted by decomposing the function in a wavelet orthonormal basis. Then the computation of the wavelet representation can be implemented with a pyramidal algorithm based on convolutions with quadrature mirror filters. Furthermore, the signal can also be reconstructed from a wavelet representation with a similar pyramidal algorithm. Practical implications of the model are illustrated. Numerous applications of multi-resolution are given in this article.

- **Spatially selective noise filtration technique based on wavelet transform**

In 1994, Xu et al. [44] described a spatially selective noise reduction method based on wavelet transform. This paper aims to develop a de-noising technique which can preserve edges while de-noising images. The edge detection is the first important step of this method. The author provides a method using direct spatial correlation to seek for sharp edges based on fact that sharp edges have large signal over many wavelet scales, and noise dies out swiftly with increasing scale. After the extraction of the position information of edges, a spatial selective filter is applied to the wavelet transform data of
the signal. Finally, reconstruction is employed to get the filtered signal. Then, this paper gives the simulations using phantom images, and real MR images to compare the results with a Wiener filter. It is found that the technique can reduce noise contents in signals and images by more than 80% while maintaining at least 80% of the value of the gradient at most edges. Furthermore, there is no Gibbs’ ringing or significant resolution loss on the filtered images. This algorithm is proved to be superior. We mention this technique is because that the preservation of edges of images can be considered to be used in pre-processing of SMI signals.

1.3 Existing problems

Based on the literature review above, some existing problems of SMI based measurement methods of $\alpha$ and pre-processing of SMI signals can be extracted.

- For those methods using SMI technique to accomplish $\alpha$ measurement, the feedback level range is limited in weak and moderate feedback regimes.

- The existing de-noising of SMI signals can change the positions of characteristic points of the signals which could result in the error of the measurement of $\alpha$.

- Nowadays the pre-processing of SMI signals and measurement of $\alpha$ using SMI technique are implemented off-line, like by Matlab, not real-time.
1.4 Aims of thesis

As for real-time high-quality sensing in industrial applications, a compact SMI system is important. That is, a small-size high-integration chip is needed to replace a PC for signal processing. Recently, digital signal processing accomplished on Field-programmable gate arrays (FPGAs) is widely used for high-speed, real-time signal processing, because FPGAs are advantageous on merging digital signal processing algorithms with other control logic. It can provide a feasible processing without scarifying accuracy or suffering extra communication latency [45]. Therefore, combining FPGA with SMI system is a good solution for the real-time sensing.

According to the existing problems, the objectives of this thesis are proposed:

1) design an SMI based $\alpha$ measurement method which is valid for a larger range of the feedback level;

2) apply pre-processing on the SMI signals to improve the signal-noise ratio (SNR);

3) implement a real-time measurement system of $\alpha$ with FPGA.

1.5 Contributions

✧ A pre-processing method combined by a wavelet transform based filter and a median filter is proposed to deal with SMI signals which can efficiently reduce noise while preserving sharp transitions in the waveform. Sharp transitions are key information for the $\alpha$ measurement in moderate and strong feedback
regimes.

✧ A novel method for expanding the range of feedback level for \( \alpha \) measurement is presented. This method is based on SMI technique, which can broaden the measurement range of \( C \) from about 3.2 to about 6.

✧ The pre-processing and the measurement method are implemented using FPGA. SMI signal is directly sent from PD into an analog-to-digital converter (ADC) on the FPGA develop board. The digital SMI signal is then processed in the FPGA chip and the result of \( \alpha \) is shown on LCD. This makes the SMI system be a compact measurement system.

1.6 Organization of thesis

This thesis consists of five chapters:

Chapter 1 introduces linewidth enhancement factor \( \alpha \) and its measurement methods. Among these methods, SMI technique with a simple experiment set-up is selected to be used in this thesis. The mathematical models and basic theory of SMI signals are reviewed after that. Then literature review on SMI based measurement methods of \( \alpha \) is given. In order to improve accuracy of measurement, pre-preprocessing of SMI signal is required. Based on the literature review, existing issues are given and purposes of the thesis are proposed.

Chapter 2 discusses the feature of SMI signals and noise included in them. Based on the features, a de-noising method combining a wavelet transform filter and a median filter is
proposed. This de-noising method can remove noise efficiently and keep sharp transitions in the waveform.

Chapter 3 firstly points out the limitation of existing $\alpha$ measurement methods and gives the reason why the method [9] is limited. Then, a measurement method of $\alpha$ is proposed which can expand the range of $C$ up to 6.

Chapter 4 states the FPGA design of the pre-processing method and measurement method of $\alpha$ for SMI signals. The FPGA design consists of ADC, pre-processing unit, measurement method and LCD display. SMI signals are sent into ADC and then processed in FPGA chips. At last, the results are shown on LCD.

Chapter 5 is the conclusion of the whole thesis and gives suggestions of future work.
Chapter 2  Pre-processing of SMI signals

Self-mixing interferometry (SMI) is useful for sensing applications. The quality of SMI signals observed in the system plays an important role in the performance of sensing. In order to retrieve the useful information carried by SMI signals for the sensing applications, high-quality SMI signals are required. However, in practice, SMI signals can be contaminated by many factors, such as ambient temperature, fluctuation in the semiconductor laser (SL) driving source and other disturbances [40]. Therefore, pre-processing of SMI signals is a necessary step before retrieving information from them.

There are several crucial characteristic points in SMI signal waveforms. The positions of these characteristic points carry information associated with the parameters of SLs [9] and the moving information of the external target [21]. These consist of sharp transitions (denoted by T), zero-crossing points (denoted by Z) and peak points (denoted by P), which are shown in Fig. 2-1.

The purpose of this chapter is to propose a de-noising method for SMI signals. Firstly, features of noise included in SMI signal waveform are analyzed and de-noising aims are
proposed. The filtering method is then described in details. This method employs a wavelet transform based filter and a median filter together. Finally, the method is verified experimentally.

![Graph of SMI signal characteristics](image)

**Figure 2-1**: Characteristics of an SMI signal under moderate feedback.

### 2.1 Noise features of SMI signals

In order to propose an efficient de-noising method for SMI signals, the features of the noise included in the waveform need to be analyzed firstly.

Two types of noises are found in SMI signals: sparkle-like and high-frequency noise. Figure 2-2 shows an SMI signal observed in our experimental set-up that can reflect the two types of noise. Sparkle-like noise can affect the location of peak points and sharp transitions; high frequency noise may lead to the error for detection of zero-crossing points. These noises would absolutely decrease the accuracy of measurement of parameters of SLs. In order to reduce the noise of SMI signals, Yu et.al [40] proposed a filtering method combining a median filter and a band-pass filter with Kaiser window
function. However, the band-pass filter with Kaiser window function can change the positions for some characteristic points such as the sharp transitions in an SMI waveform.

Another method, a neural network interpolation technique for the noise elimination of SMI signals was presented by Wei et al. [41]. But this method only covers the weak and moderate optical feedback regimes and also has the same problems with the method in [40].

![Figure 2-2: An experimental SMI signal.](image)

In accordance with the noise and waveform features of an SMI signal, the filtering requirements are determined as follows:

(a) Sparkle-like noise should be removed without changing the locations of it and its surrounding points.

(b) The positions of sharp transitions should be kept unchanged after filtering.

(c) High-frequency noise should be reduced.
2.2 Filter design

In 2007, Yu et.al [40] indicated that the usual moving average filters could not work well for SMI signals and presented a filtering method that combined a median filter and a band-pass filter based on Kaiser window function. The median filter deals with sparkle-like noise very well but is poor for high-frequency noise. The Kaiser window function based band-pass filter can improve performance in reducing high-frequency noise. Unfortunately, it can meanwhile change the waveform especially for sharp transitions. To solve this problem, a wavelet transform based filter is employed to reduce high-frequency noise while preserving original waveform instead of the band-pass filter. A median filter is used to remove sparkle-like noise, which is introduced afterwards.

2.2.1 Wavelet transform based filtering process

Wavelet transform is thought as “mathematical microscope” on account of its fine frequency property, which is appropriate to the processing of SMI signals. A wavelet transform based filter can reduce noise effectively with little resolution loss and most sharp transitions can be preserved [46-48]. This provides a powerful tool for the processing of SMI signals. Wavelet transform represents signals in multi-resolution, and decomposes signals into different scales [43, 49, 50]. Sharp transitions are preserved and described very well in wavelet expansions [43]. Most noise is restricted to small scales, so the filtering process at small scales decreases the noise preferentially [44]. Before
applying wavelet transforms, the information on the sharp transitions should be acquired and preserved during the filtration. In 1994, Xu et al. [44] proposed an easy way to locate sharp transitions for images or signals using direct correlation between signals at different scales. This method is efficient for most signals. For SMI signals, however, there is a better way to locate the transitions. The difference in the amplitude between two points in a sharp transition is significant, so differential is better to locate the positions of the transitions. In the following part of this section, the basic theory of wavelet transform is introduced first and the de-noising method is then described in detail.

### 2.2.1.1 Wavelet theory

Generating a set of filters by dilation and translation of a generating wavelet is the base of wavelet theory. All of the wavelets are scaled versions of the “mother wavelet”, which means that only one filter should be designed and others can use the same scaling rules both in time and frequency domain [51].

Families of wavelets are generated from the mother wavelet \( \Psi(t) \):

\[
\Psi_{a,b}(t) = \frac{1}{\sqrt{a}} \Psi\left(\frac{t-b}{a}\right)
\]

where \( a \) is the dilation (scale) parameter, and \( b \) is the translation parameter. The wavelets are contracted \( a < 1 \) or dilated \( a > 1 \) and are moved over the signal to be analyzed by time step \( b \) (real value). Contraction and dilation scales the frequency
response to the required frequency band. A set of wavelets can be seen as a filter bank.

For admissibility as a wavelet the following condition has to be met:

\[
\int \left| \hat{\Psi}(w) \right|^2 \frac{dw}{|w|} < \infty
\]  \hspace{1cm} (2.2)

which means that if the wavelet is differentiable then:

\[
\hat{\Psi}(0) = 0
\]  \hspace{1cm} (2.3)

The continuous wavelet transform (CWT) is defined as:

\[
CWT(b, a) = \frac{1}{\sqrt{a}} \int s(t) \Psi\left(\frac{t-b}{a}\right) dt
\]  \hspace{1cm} (2.4)

The discrete wavelet transform (DWT) is given by:

\[
DWT(a^i, a'^i) = \frac{1}{\sqrt{a'}} \sum_k \Psi\left(\frac{k}{a'} - n\right) s(k)
\]  \hspace{1cm} (2.5)

where \( i \) is an integer. When \( a = 2 \), the DWT calculates data points at a dyadic grid.

\section{2.2.1.2 Filtering method}

The filtering method based on wavelet transform for SMI signals is as depicted in Fig. 2-3.

The symbols in Fig. 2-3 are summarized in Table 2-1. The filtering procedure is described as below. The overall filtering process includes four stages. The first stage is discrete wavelet transform (DWT); the second one is single level reconstruction; the third one, locating sharp transitions and the last is reconstruction. For convenience of description, a noisy SMI signal is used to help demonstrate the method.
Figure 2-3: Flow chart of the wavelet transform based filtering process.

Table 2-1: Meanings of symbols in Fig. 2-3.

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$g[n]$</td>
<td>A sampled SMI signal.</td>
</tr>
<tr>
<td>$g'[n]$</td>
<td>The differential of $g[n]$ which helps detecting the transitions.</td>
</tr>
<tr>
<td>high-pass filter</td>
<td>Decomposition filter.</td>
</tr>
<tr>
<td>low-pass filter</td>
<td>Decomposition filter.</td>
</tr>
<tr>
<td>high-pass_r filter</td>
<td>Reconstruction filter.</td>
</tr>
<tr>
<td>low-pass_r filter</td>
<td>Reconstruction filter.</td>
</tr>
<tr>
<td>up sampling block</td>
<td>Up sampling factor is 2.</td>
</tr>
<tr>
<td>down sampling block</td>
<td>Down sampling factor is 2.</td>
</tr>
<tr>
<td>$d[k]$</td>
<td>High frequency coefficients.</td>
</tr>
<tr>
<td>$d[k]$</td>
<td>Low frequency coefficients</td>
</tr>
<tr>
<td>$d[n]$</td>
<td>Reconstructed high frequency component.</td>
</tr>
<tr>
<td>$a[n]$</td>
<td>Reconstructed low frequency component.</td>
</tr>
<tr>
<td>$d_r[n]$</td>
<td>Filtered high frequency component.</td>
</tr>
<tr>
<td>$g[n]$</td>
<td>Filtered SMI signal.</td>
</tr>
</tbody>
</table>

Simulation SMI signals without and with noise are employed to help demonstrate this method in detail. Fig. 2-4 (a) shows a simulation SMI signal with $C = 3$ and $\alpha = 3$. The
frequency of the SMI signal is 71 Hz and the sampling frequency is 102.4 KHz. (b) shows the signal with noise of signal to noise ratio (SNR) of 20 dB.

Figure 2-4: (a) A simulation SMI signal with $C = 3$ and $\alpha = 3$; (b) simulation signal with noise at SNR of 20 dB.

Step1, DWT, is applied on an SMI signal to decompose it into different frequency contents. DWT is the signal passes a series of low- and high-pass filters. The low-pass filters produce low-frequency coefficients of the signal (denoted by $a[k]$), while the high-pass filters give high frequency coefficients (denoted by $d[k]$). Down-sampling is needed after filtering the signal through the analysis filter bank because a single-length input is being converted to a double length output. This operation is defined by the following equations [52]:

$$d[k] = \sum_{n=0}^{N-1} g[2k - n]H[n]$$  \hspace{1cm} (2.6)

$$a[k] = \sum_{n=0}^{N-1} g[2k - n]L[n]$$  \hspace{1cm} (2.7)
In this pair of equations, \( N \) is the number of the filter taps and \( g \) denotes the sampled self-mixing signal, \( H[k] \) represents the high-pass filter and \( L[k] \) indicates the low-pass filter. After decomposition, \( d[k] \) and \( a[k] \) are obtained. In this study, Daubechies 4-tap (that is \( N=4 \)) wavelet is chosen as the mother wavelet because it has the greatest similarity to the fringe shape of SMI signals, and therefore resulting in the best performance. The shape of this wavelet base is shown in Fig. 2-5, and the corresponding FIR filters’ coefficients are: low-pass filter \( L[k] \) \{0.1294 0.2241 0.8365 0.4830\}; high-pass filter \( H[k] \) \{-0.4830 0.8365 -0.2241 -0.1294\} [53].

![Figure 2-5: Daubechies 4-tap wavelet base and scaling base.](image)

Step 2, reconstruction, is applied to each frequency component at different levels. The process is accomplished by passing each frequency component into the reconstruction filters and the up-sampling blocks. Then high-frequency \( d[n] \) and low-frequency \( a[n] \) content are derived. The reconstruction FIR filters’ coefficients are listed as following: low-pass reconstruction filter: \{0.4830 0.8365 0.2241 -0.1294\}; high-pass reconstruction filter:
Figure 2-6 gives the reconstruction results of different frequency content. Figure 2-6(a) shows the reconstructed low frequency content. Figure 2-6(b) gives the reconstructed high frequency content at first level, and high frequency content at second level is shown in Fig. 2-6(c). After the reconstruction, the reconstructed low frequency content is kept because most frequency content of the signal is included in this part, while the high frequency component is processed using the position information obtained in the next step. That is because the high frequency noise and the sharp transitions of the signal are contained in the high frequency content [44].

Figure 2-6: (a) Low frequency content of the SMI signal; (b) high frequency content of the SMI signal at first level; (c) high frequency content of the SMI signal at second level.

Step 3, the key step of the processing, is to locate the position of the sharp transitions. Xu’s method [44] is easy and appropriate for most signals. However, because of the characteristics of SMI signals, differential operation is simpler and more precise. A spatially selective filter, \( f[n] \), is then applied to the reconstructed high frequency
content with the position information of the transitions; it spatially selects which part of the data (the transitions) to preserve and which part (noise) to eliminate. In Figure 2-7, the illumination of the spatial filter is given. The derivative of the simulation SMI signal is shown in (a). Figure 2-7(b) gives the spatial filter based on the information from the derivative of the SMI signal. Figure 2-7(c) is the filtered high frequency content at first level scale.

![Figure 2-7](image)

Figure 2-7: (a) Derivative of the simulation SMI signal; (b) the spatial filter based on the information from the derivative of the SMI signal; (c) filtered high frequency content at first level scale.

In step 4, the processed high frequency components $d_f[n]$ are combined with the reconstructed low-frequency components $a[n]$ to reconstruct the filtered signal. Figures 2-8(a) and (b) show the simulation signal without and with noise, respectively. The final filtered results are illustrated in Fig. 2-8(c), in which it can be seen that the high-frequency noise is reduced. After the wavelet filtering, most sharp edges are preserved and some of them are even enhanced [46-48]. The latter parts need to be
fixed. Thus, another median filter is utilized to process the data. Figure 2-8(d) shows the enlarged transitions part before and after the filtering process. In this figure, it can be seen that the positions of sharp transitions are preserved. The noisy SMI is thus processed well.

Figure 2-8: Filtering results of a simulation noisy SMI signal: (a) a simulation pure SMI signal; (b) the SMI signal with noise (SNR =20dB); (c) filtered SMI signal; (d) an enlarged transitions parts before (left subfigure) and after (right subfigure) the filtering.

### 2.2.2 Median filter

In the previous section, a wavelet transform filter was proposed to remove high-frequency noise while preserving sharp transitions. For another type of noise, sparkle-like noise, a median filter is employed to deal with it based on the work in [40]. Median filters are widely used in image processing because they have two intrinsic properties: edge preservation and efficient noise attenuation, with robustness against
impulsive-type noise [54]. To compute the output of a median filter for 1-D signal, an odd number of sample values are sorted, and the middle or median value is used as the filter output. If the filter length is \( N = 2K + 1 \), the filtering procedure is denoted as

\[
Y(n) = MED[X(n-K),...,X(n),...,X(n+K)]
\]

(2.8)

where \( X(n) \) and \( Y(n) \) are the \( n_{th} \) sample of the input and output sequences, respectively. The value of \( Y(n) \) is determined as the \( (K+1)_{th} \) value of the sorted sequence. For example, let the input be \( X(n)=[3,2,125,8,9] \) and the filter length is \( N = 3 \). Thus each sample of output sequence can be calculated as follows:

\[
Y(1) = MED[3,3,2] = MED[2,3,3] = 3
\]

\[
Y(2) = MED[3,2,125] = MED[2,3,125] = 3
\]

\[
Y(3) = MED[2,125,8] = MED[2,8,125] = 8
\]

\[
Y(4) = MED[125,8,9] = MED[8,9,125] = 9
\]

\[
Y(5) = MED[8,9,9] = MED[8,9,9] = 9
\]

It should be noted that the first and last values of the input are repeated because they are the boundaries. Then the output sequence is \( Y(n)=[3,3,8,9,9] \). From the comparison of \( X(n) \) and \( Y(n) \), it can be seen that \( X(3)=125 \), which is much larger than other input samples, is removed. \( X(3)=125 \) can be considered as a sparkle-like noise.

For SMI signals in moderate and strong feedback regimes, sparkle-like noise will distort SMI signals and decrease the accuracy of sensing applications, such as parameter measurement and displacement [40]. Several enlarged sparkle-like noises of the noisy
experimental signal in Fig. 2-2 are shown in Fig. 2-9.

Figure 2-9: Enlarged sparkle-like noise of the SMI signal shown in Fig. 2-6.

From numerous SMI signals obtained from our experimental set-up, the length of most sparkles is no more than 5 samples. Based on this length, we pick 5, 10 and 20 as the median filter lengths, and test their performance. Figures 2-10 (a)-(g) give an experimental, filtered SMI signal with filter lengths of 5, 10 and 20, respectively.
Figure 2-10: Comparison of performances of median filters with different filter lengths: (a) experimental signal; (b) filtered signal when N=5; (c) enlarged transition of the signal in (b); (d) filtered signal when N=10; (e) enlarged transition of the signal in (d); (f) filtered signal when N=20; (g) enlarged transition of the signal in (f).
From the figures above, it can be seen that the sharp transitions are preserved while filter length $N = 5$ but the high-frequency noise still needs to be reduced. When $N = 10$, the waveform of the SMI signal is good but the sharp transitions have already changed, as can be seen in the zoomed figure. When $N = 20$, the waveform seems very well but distortions even appear in the filtered signal. So a median filter with a smaller length preserves sharp transitions more precisely but cannot reduce high-frequency noise efficiently, while a median filter with larger length can change the sharp transitions. So a 5-sample median filter is chosen to remove sparkle-like noise and the high frequency noise can be reduced by the wavelet transform filter described in the previous section. The combination of these two filters gives reliable performance to SMI signals, both types of being dealt with. Experimental verification is given in next section.

### 2.3 Performance test

In this section, the filtering method is applied to experimental signals to verify its feasibility. The SMI experimental set-up is shown in Fig. 2-11. The core part of this system consists of an SL, a lens and a target. The SL works when the temperature is 25 °C controlled by a thermoelectric temperature controller. An SL controller also controls the SL and the range of injection current is set from 70 mA to 90 mA. The target is a loudspeaker driven by a signal generator, and the driving signal’s frequency range is from 60Hz to 71Hz. When all the components in the system work, the SMI signals can be detected from the photo diode (PD) and sent into a PC.
The temperature controller used is a thermoelectric temperature controller TED200 provided by Thorlabs GmbH. TED200 is an extremely precise temperature controller for laser diodes and detectors [55].

![Diagram](image)

Figure 2-11: Experimental set-up of the SMI system

The TED 200 works excellently for:

- wavelength stabilization of laser diodes
- noise reduction of detectors
- wavelength tuning by regulating the temperature
- modulation of wavelength by tuning the temperature

The SL controller used, an LDC2000, is also provided by Thorlabs GmbH. Thorlabs’ LDC2000 laser diode controller is an extremely precise controller for SLs. When it is used with the Thorlabs TEC2000 temperature controller, the laser current and the optical output power and the temperature of the connected SL can be precisely regulated. Thus the SL works as a stable optical transmitter [14].

LDC2000 is suitable for:

- safe and simple operation of all laser diodes up to 2A
- power stabilized light sources
- wavelength tuning by controlling the current
- wavelength modulation by current modulation

In this project, HL7851G is chosen as our SL. HL7851G is a high power 0.78 μm band GaAlAs laser diode with a multi-quantum well (MQW) structure. It is suitable as a light source for optical disk memories, levelers and various other types of optical equipment. Hermetic sealing of the package assures high reliability [56].

The absolute maximum ratings (T=25 °C) and optical and electrical characteristics (T = 25 ±3 °C) of HL7851G are listed in Tables 2-2 and 2-3, respectively.

| Table 2-2: Absolute maximum ratings (T=25 °C) of HL7851G [56]. |
|-----------------|-----------------|-----------------|---|
| Item            | Symbol          | Rated Value     | Unit |
| Optical output power | $P_o$           | 50              | mW   |
| Pulse optical output power | $P_{(pulse)}$   | 60              | mW   |
| SL reverse voltage | $V_{R(LD)}$     | 2               | V    |
| PD reverse voltage | $V_{R(PD)}$     | 30              | V    |
| Operating temperature | $T_{opr}$       | -10 to +60      | °C   |
| Storage temperature | $T_{stg}$       | -40 to +85      | °C   |
Table 2-3: Electrical characteristics ($T = 25 \degree C$) of HL7851G [56].

<table>
<thead>
<tr>
<th>Items</th>
<th>Symbol</th>
<th>Min</th>
<th>Typ</th>
<th>Max</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optical output power</td>
<td>$P_0$</td>
<td>50</td>
<td></td>
<td></td>
<td>mW</td>
</tr>
<tr>
<td>Threshold current</td>
<td>$I_{th}$</td>
<td></td>
<td>45</td>
<td>70</td>
<td>mA</td>
</tr>
<tr>
<td>Slope efficiency</td>
<td>$\eta$</td>
<td>0.35</td>
<td>0.55</td>
<td>0.7</td>
<td>mW /mA</td>
</tr>
<tr>
<td>Operating current</td>
<td>$L_{op}$</td>
<td></td>
<td>140</td>
<td>170</td>
<td>mA</td>
</tr>
<tr>
<td>SL Operating voltage</td>
<td>$V_{op}$</td>
<td></td>
<td>2.3</td>
<td>2.7</td>
<td>V</td>
</tr>
<tr>
<td>Lasing wavelength</td>
<td>$\lambda_p$</td>
<td>775</td>
<td>785</td>
<td>795</td>
<td>nm</td>
</tr>
<tr>
<td>Beam divergence (parallel)</td>
<td>$\theta / /$</td>
<td>8</td>
<td>9.5</td>
<td>12</td>
<td>deg.</td>
</tr>
<tr>
<td>Beam divergence</td>
<td>$\theta$</td>
<td>18</td>
<td>23</td>
<td>28</td>
<td>deg.</td>
</tr>
<tr>
<td>Monitor current</td>
<td>$L_s$</td>
<td>25</td>
<td></td>
<td>150</td>
<td>A</td>
</tr>
<tr>
<td>Astigmatism</td>
<td>$A_s$</td>
<td></td>
<td>5</td>
<td></td>
<td>m</td>
</tr>
</tbody>
</table>

The proposed filtering method is applied on experimental signals to verify its correctness.

The SMI signal retrieved from the PD is shown in Fig. 2-12(a). Figure 2-12(b) illustrates the processed signal and an enlarged transition is shown in Fig. 2-12(c). It can be seen clearly that sparkle-noise is removed and high-frequency noise is reduced. It can also be seen from the enlarged transition that the sparkle-like noise is removed and the information about the transitions is preserved.

Figure 2-12: Filtering results of an experimental SMI signal: (a) original experimental SMI signal; (b) filtered SMI signal; (c) enlarged transition before(left subfigure) and after(right subfigure) filtering.
2.4 Summary

This chapter firstly draws attention to characteristic points which include important information for measuring the parameters of SLs. The features of the noise included in the SMI signal waveform are then analyzed. Based on the analysis of characteristic points and noise, a de-noising method combing a wavelet transform filter and a median filter is proposed. A median filter can efficiently remove sparkle-like noise but cannot reduce the high-frequency noise. A wavelet transform filter can efficiently reduce high-frequency noise while preserving the sharp transitions in the waveform. The combination of these two methods gives good performance for de-noising SMI signals. Both simulation and experimental data are employed to verify this method.
Chapter 3 Improved algorithm on $\alpha$ measurement

As outlined in the Chapter 1 literature review, measurement of the linewidth enhancement factor ($\alpha$) has been researched extensively [1, 3-5, 8-11, 17, 19, 57-59]. Among these methods, SMI technology has advantages over other traditional interferometries, such as simplicity of system structure and ease in system calibration. However, existing measurement methods have their own drawbacks. The most important drawback is that they are limited to a narrow range of feedback level. The methods described in [8, 10, 39] are limited in the range (0~1) of feedback level factor ($C$) and the method set out in [9] works when $1 < C < 3.6$. This chapter will describe a method that can expand the measurement range up to $C = 6$. Section 3.1 gives the analysis of SMI signal waveform. Based on the relationships between the parameters of semiconductor lasers (SLs) and the time intervals in waveforms, the proposed measurement method is described in Section 3.2. Section 3.3 and Section 3.4 present the simulation and experiment results, respectively.
3.1 Analysis of SMI signal waveform

SMI signals observed in an SMI system usually have a fringe structure similar to traditional interference fringes, and each fringe period corresponds to a half wavelength displacement of the external cavity in a weak or moderate feedback regime. When $C > 1$, as the result of mode hopping, hysteresis phenomenon occurs, significantly changing the fringe shapes \[9, 22, 36\]. Using Equations (1.25)-(1.27), an example of the hysteresis in an SMI waveform is shown in Fig. 3-1, where $C = 2$ and $\alpha = 5$. When $\phi_0$ increases, $\phi_F$ and $g(\phi_0)$ follow the path $A_1 - B - B_1$, but when it decreases, they will follow the path $B_1 - A - A_1$. The area $A_1 - B - B_1 - A$ is called hysteresis area. $\phi_{0, AB}$ is the width of the hysteresis area. For ease of description, “up fringes” is used to denote the fringe segments of $g(\phi_0)$ corresponding to increasing $\phi_0$, and “down fringes” stands for the fringe segments corresponding to decreasing $\phi_0$. Also note that there are a few characteristic points in Fig. 3-1, including zero-crossing points of $g(\phi_0)$ where $\phi_0 = \phi_{0, C}$ and $\phi_0 = \phi_{0, D}$, and jumping points of $g(\phi_0)$ where $\phi_0 = \phi_{0, A}$ and $\phi_0 = \phi_{0, B}$. When using the approach in \[9\], $\phi_{0, AB} = \phi_{0, B} - \phi_{0, C}$ on an up fringe and $\phi_{0, AD} = \phi_{0, D} - \phi_{0, A}$ on a down fringe must be measured. Obviously, the approach in \[9\] requires the existence of a zero-crossing point $\phi_{0, C}$ in the up fringe segment and a zero-crossing point $\phi_{0, D}$ in the down fringe segment. However, these two zero-crossing points may not exist in some cases. When $C$ increases, the hysteresis area becomes wider and the fringes are declined to the right more right, making the jumping point $\phi_{0, B}$ shift to the
right, and the zero-crossing point on up fringes will disappear when $\phi_{0,B}$ is on the right side of $\phi_{0,D}$. Meanwhile, when $C$ increases, the jumping line $\phi_{0,A}$ in Fig. 3-1 will move leftward, resulting in a loss of zero-crossing points on the down fringes if it is on the left side of $\phi_{0,C}$. Hence the approach in [9] is only valid for a narrow range of $C$.

![Figure 3-1: Hysteresis phenomenon of an SMI signal, where $C = 2$ and $\alpha = 5$ [60].](image)

### 3.2 Algorithm description

According to [9], the hysteresis width can be expressed as follows:

\[
\phi_{0,CB} = \sqrt{C^2 - 1} + \frac{C}{\sqrt{1 + \alpha^2}} + \arccos\left(-\frac{1}{C}\right) - \arctan(\alpha) + \frac{\pi}{2}
\]

\[
\phi_{0,AD} = \sqrt{C^2 - 1} - \frac{C}{\sqrt{1 + \alpha^2}} + \arccos\left(-\frac{1}{C}\right) + \arctan(\alpha) - \frac{\pi}{2}
\]

where $\phi_{0,CB}$, $\phi_{0,AD}$ can be determined from time intervals between the characteristic points in the waveform of SMI signals. Figure 3-2 gives a demonstration of their measurement. $\phi_{0,CB} = 2\pi \times t_{13} / T_1$ and $\phi_{0,AD} = 2\pi \times t_{24} / T_2$, where $t_{CB}$, $T_1$, $t_{AD}$ and $T_2$ are labeled in Fig. 3-2.
Figure 3-2: An SMI signal with $C = 2$ and $\alpha = 5$, used to show time intervals $t_{\text{CB}}$, $T_1$, $t_{\text{AD}}$ and $T_2$.

When $C$ increases, the valley points of the up fringes in the SMI signal waveform rise up and the tops of the down fringes descend gradually. The zero-crossing points of the up fringes disappear first. If $C$ continues increasing, the zero-crossing points of the down fringes disappear. The change is show step by step in Fig. 3-3. Figure 3-3(a) shows an SMI signal with $C = 3$, $\alpha = 5$; (b) gives an SMI signal with $C = 4$, $\alpha = 5$; a signal with $C = 5$, $\alpha = 5$ and another with $C = 6$, $\alpha = 5$ are given in (c) and (d), respectively. These figures show the trend of the change of the fringe pattern in an SMI signal’s waveform.

When zero-crossing points in up fringes do not exist, $t_{13}$ disappear, which means $\phi_{0,\text{CB}}$ disappear. Then, the Eq. (3.1) cannot be used. Thus, the measurement method described in [9] does not work.
Figure 3-3: Trend of change in SMI signal waveform: (a) an SMI signal with \( C = 3 \) and \( \alpha = 5 \); (b) an SMI signal with \( C = 4 \) and \( \alpha = 5 \); (c) an SMI signal with \( C = 5 \) and \( \alpha = 5 \); (d) an SMI signal with \( C = 6 \) and \( \alpha = 5 \).

However, \( \phi_{0,AD} \) in the down fringe exists in a larger range of \( C \) than \( \phi_{0,CB} \) in the up fringes, which means that Eq. (3.2) can work in a larger feedback level. In this equation, \( \phi_{0,AD} \) can be obtained from the waveform of SMI signals. Only two variables, \( C \) and \( \alpha \), are unknown. As for \( C \), it can be measured by using the method in [60]. Then only one unknown variable, \( \alpha \), is left in Eq. (3.2), and can be solved. Firstly, a review on the
$C$ estimate method is given as below.

This method analyzes the Lang-Kobayashi (L-K) model and takes the Fourier transform of the phase content in Eq. (1.25). Then a rough estimate of the spectrum of external light phases of a free-running SL is made to find where it can be seen as zero. The strategies are shown as follows: 1) pre-processing of an SMI signal including de-noising and normalization; 2) determining the vibration frequency by auto-correlation [10]; 3) carrying out arccosine function operation on $g(t)$ and phase-unwrapping operation to obtain external light phases of an SL with feedback; 4) applying Fast Fourier Transform (FFT) on phase contents and acquiring their spectra; 5) calculating $C$ value in a special frequency domain. The significant advantage of this method is that it can cover all the feedback levels for $C$ estimation.

After $C$ is estimated, $\alpha$ can be calculated by solving Eq. (3.2). In this method, the $\alpha$ measurement range of $C$ can be expanded up to $C = 6$, which will be explained in the following part.

For different $\alpha$, the measurement range also varies. In the literature [9], the range of $C$ is limited as $C < 3.2$ by $\phi_{0,CB}$ because only when $\phi_{0,CB} < 2\pi$, can Eq. (3.1) be used for measurement of parameters. This boundary can be obtained from the relationship between $\phi_{0,CB}$ and $t_{13}$. Because $\phi_{0,CB} = 2\pi \times t_{13} / T_1$, where $t_{13}$ is the interval of zero-crossing points and $T_1$ is the interval of the corresponding up fringe, $t_{13}$ is always less than $T_1$, which results in $\phi_{0,CB} < 2\pi$. This situation is also suitable for $\phi_{0,AD}$. 

54
Thus, when $\phi_{0,\text{CB}}=2\pi$, the boundary of $C$ in method [9] can be obtained. However, in the proposed method, the measurement range is not limited by $\phi_{0,\text{CB}}$. Instead, it is only determined by the value of $C$, which makes the zero-crossing points in down fringes disappear, meanwhile, $\phi_{0,\text{AD}}$ decreases to zero. Since $\phi_{0,\text{AD}}$ disappears much later than $\phi_{0,\text{CB}}$, the range of measurement is broadened.

In order to determine the measurement ranges of $C$, different values of $\alpha$ are pre-set and $\phi_{0,\text{AD}}$ is set at its boundary value, $2\pi$. By solving Eq. (3.2) with the pre-set $\alpha$ and $\phi_{0,\text{AD}}$, a broadened range of $C$ are acquired, as shown in Table 3-1. Also the measurement range of the method in [9] is listed in Table 3-1. Table 3-1 shows that the proposed method is able to measure $\alpha$ over a wider range of $C$.

Table 3-1: Comparison of $C$ range between the proposed method and the method [9].

<table>
<thead>
<tr>
<th>$C$ range</th>
<th>$\alpha$</th>
<th>$\alpha = 3$</th>
<th>$\alpha = 4$</th>
<th>$\alpha = 5$</th>
<th>$\alpha = 6$</th>
<th>$\alpha = 7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>the proposed method</td>
<td>$1 &lt; C &lt; 7.2$</td>
<td>$1 &lt; C &lt; 6.4$</td>
<td>$1 &lt; C &lt; 6.0$</td>
<td>$1 &lt; C &lt; 5.7$</td>
<td>$1 &lt; C &lt; 5.5$</td>
<td></td>
</tr>
<tr>
<td>the method in [9]</td>
<td>$1 &lt; C &lt; 3.2$</td>
<td>$1 &lt; C &lt; 3.4$</td>
<td>$1 &lt; C &lt; 3.6$</td>
<td>$1 &lt; C &lt; 3.7$</td>
<td>$1 &lt; C &lt; 3.8$</td>
<td></td>
</tr>
</tbody>
</table>

Since the estimation method of $C$ [60] is employed in this proposed method, the details of it are worth noticing:

Taking Fourier transform on both sides of Eq. (1.25), the following equation is obtained.

$$
\Phi_F(f) = \Phi_0(f) - C \cdot F\{\sin[\phi_F(n)] + \arctan(\alpha)\}
$$

(3.3)

where $F\{\cdot\}$ denotes the operation of Fourier Transform, while $\Phi_F(f)$ and $\Phi_0(f)$ are the Fourier transforms of $\phi_F(n)$ and $\phi_0(n)$ respectively. A function is introduced
for ease of explanation:

\[ \phi_1(n) = \sin[\phi_x(n) + \arctan(\alpha)] \]  \hspace{1cm} (3.4)

So Eq. (3.3) becomes

\[ \Phi_x(f) = \Phi_0(f) - C \cdot F\{\phi_1(n)\} = \Phi_0(f) - C \cdot \Phi_1(f) \]  \hspace{1cm} (3.5)

where \( \Phi_1(f) \) is the Fourier transforms of \( \phi_1(n) \). \( \phi_x(n) \) and \( \Phi_x(f) \) can be obtained from Eq. (1.27) using the method [21]. \( \Phi_1(f) \) can be calculated by Eq. (3.4).

In addition, Yu et al. gives some area in the spectrum where \( \Phi_x(f) \) and \( \Phi_1(f) \) are non-zero while \( \Phi_0(f) = 0 \). Therefore \( C \) can be calculated:

\[ C = \left| \frac{\Phi_x(f)}{\Phi_1(f)} \right| \]  \hspace{1cm} (3.6)

In this method, there is an assumption that can produce estimation error. In Eq. (3.4), \( \phi_1(n) \) is calculated with \( \phi_x(n) \) and \( \alpha \). Because \( \alpha \) is unknown, a common used values is assumed, such as \( \hat{\alpha} = 3 \). Therefore, an error arises here because of this assumed \( \hat{\alpha} \). Although there is an small error when calculating \( C \), \( \alpha \) obtained using the proposed method is usually closer to the true value than the assumed value. Thus, this measured \( \alpha \) is used to replace the previous \( \hat{\alpha} \) and the new \( \hat{\alpha} \) is substituted into Eq. (3.4). This process of calculation-replacement improves the accuracy of the measurement. The terminal condition of the loop is that it repeats until that the measured \( \alpha \) equals to the previous \( \hat{\alpha} \). A flow chart of this process is shown in Fig. 3-4.
3.3 Simulation Verification

In order to verify the feasibility of the proposed method, computer simulation is applied to generate SMI signals with different values of $\alpha$ and $C$ using Eq. (1.25)-(1.28).

Considering that the moving target vibrates in a simple harmonic mode, $L(n)$ in Eq. (1.28) is a sinusoid function. Then $\phi_0(n)$ can be given as below:

$$\phi_0(n) = \varphi_0 + \Delta \varphi \sin(2\pi f_0 t)$$  \hspace{1cm} (3.7)

Figure 3-4: Flow chart of the proposed method.
where $\phi_0$ is the light phase at the equilibrium position of a vibrating target, $\Delta\phi$ is the maximum phase deviation caused by target vibration amplitude, $f_0$ is the vibration frequency. Main parameters of simulations signals are as following: $\phi_0=3.9 \times 10^6$ (rad), $\Delta\phi=7\pi$, $f_0=71$ Hz and sampling frequency is $f_s=102400$ Hz.

Different values of $C$ and $\alpha$ are used to generate different SMI signal to test the method. Here, a simulation signal with $\alpha=5$ and $C=4$ is generated to demonstrate the simulation. From Eq. (1.25)-(1.27), $\phi_0(n)$, $g(n)$ and $\phi_\ell(n)$ are obtained and shown in Fig. 3-5(a), (b) and (c). The calculated $\phi_1(n)$ is shown in Fig. 3-5(d).

After the simulation signals are obtained, $C$ is measured firstly using the method in [60]. When using this method, a common pre-set $\hat{\alpha}=3$ is used to calculate $\phi_1(n)$ in Eq. (3.4). By using this method, $C$ is estimated as 4.04. Then the time intervals $t_2$ and $T_2$ in $g(n)$ are measured and $\alpha$ is calculated with Eq. (3.2). The value of measured $\alpha$ is 5.07. Then this measured $\alpha$ replaces $\hat{\alpha}$ to be substituted into Eq. (3.6) and the following step is operated again. This calculate-replace loop stops when $\alpha=4.97$ and $C=3.99$ and they are unchanged even the loop operates once more.

The ultimate results are more accurate than the initial ones.

Different simulation SMI signals with different values of $C$ and $\alpha$ are applied to verify this method. According to limited space, only the simulation results are given in Fig. 3-6.
Figure 3-5: Waveforms of $g(n)$ and $\phi_n(n)$ where $\alpha=5$ and $C=4$: (a): $g(n)$; (b): $\phi_0(n)$; (c): $\phi_F(n)$; (d): $\phi_1(n)$.

X axis stands for different simulation values of $C : 3, 4, 5, 6$ and Y axis for values of measured $\alpha$ corresponding to different $C$ and $\alpha$. The dashed line gives the true
value of $\alpha$. From this figure, it can be seen that the proposed method is able to work out $\alpha$ over a larger range of feedback level.

![Figure 3-6: Simulation results of the proposed method.](image)

### 3.4 Experimental Verification

The experimental set-up is already described in Section 2.3. SMI signals are retrieved from the PD in the system. One experimental signal is shown in Fig. 3-7, which has been already processed using the method described in the previous chapter, and this signal is used as a demonstration of experiment process.

![Figure 3-7: A processed experimental signal.](image)
The recovered $\phi_e$, $\phi_1$ and their spectrums are shown in Fig. 3-8. Using the method in [60], the value of $C$ is estimated as 3.2. The next step is to measure $\tau_{24}$ and $T_2$ referring to Fig. 3-2, and to calculate the value of $\alpha$ according to Eq. (3.2). The measurement result is $\alpha=5.5$.

Figure 3-8: (a) Recovered $\phi_e$ and its spectrum; (b) $\phi_1$ and its spectrum.

Several experimental SMI signals are employed to verify the proposed method.

Furthermore, the method described in [9] is applied to test the consistence between the
two method. The measurement results are shown in Table 3-4, in which a comparison between the proposed method and the method in [9] is given.

Table 3-2: Comparison of measurement results of proposed method with method in [9].

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$C &lt; 3$</td>
<td>1.3</td>
<td>3.2</td>
<td>1.3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>1.4</td>
<td>2.9</td>
<td>1.7</td>
<td>2.6</td>
</tr>
<tr>
<td></td>
<td>1.9</td>
<td>2.9</td>
<td>2</td>
<td>2.7</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2.8</td>
<td>2.4</td>
<td>2.5</td>
</tr>
<tr>
<td></td>
<td>2.1</td>
<td>3.1</td>
<td>2.2</td>
<td>2.8</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2.8</td>
<td>2.3</td>
<td>2.5</td>
</tr>
<tr>
<td></td>
<td>2.3</td>
<td>3.1</td>
<td>2.4</td>
<td>3.7</td>
</tr>
<tr>
<td>$C &gt; 3$</td>
<td>N/A</td>
<td>N/A</td>
<td>3.2</td>
<td>5.5</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
<td>N/A</td>
<td>3.3</td>
<td>2.3</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
<td>N/A</td>
<td>3.6</td>
<td>3.6</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
<td>N/A</td>
<td>4</td>
<td>3.5</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
<td>N/A</td>
<td>4.4</td>
<td>3.2</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
<td>N/A</td>
<td>4.7</td>
<td>4.9</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
<td>N/A</td>
<td>5.4</td>
<td>3.2</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
<td>N/A</td>
<td>6.2</td>
<td>3.1</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
<td>N/A</td>
<td>6.4</td>
<td>2.7</td>
</tr>
</tbody>
</table>

From the above table, it can be seen that when $C < 3$, the results of proposed method show good agreements with the method in [9]. When $C > 3$, the method in [9] will not work. Conversely, the proposed method gives a reliable measurement.

### 3.5 Summary

This chapter first analyzes the hysteresis area and characteristic points, jumping points and zero-crossing points, in an SMI signal waveform. On the basis of this analysis, the reason why the method described in [9] only covers a narrow range of $C$ is explained.
Based on the relationships between time intervals in SMI signals waveform and parameters, a method which can broaden the range of $C$ for the measurement of $\alpha$ is proposed. Both simulation and experiment are applied to verify this method and the results show that it is a reliable method.
Chapter 4  Real-time system

The previous chapters have introduced the SMI sensing system. The pre-processing algorithm of SMI signals and the SMI based $\alpha$ measurement method are accomplished off-line, using Matlab, not in real-time.

For real-time high-quality sensing in industrial applications, a compact SMI system is required. That is to say, a small-size high-integration chip is needed to replace a PC for signal processing. In recent years, digital signal processing accomplished on field-programmable gate arrays (FPGAs) has been widely used for high-speed, real-time signal processing, because FPGAs offer advantages in merging digital signal processing algorithms with other control logic. They can provide a feasible processing without sacrificing accuracy or suffering extra communication latency [61]. Combining FPGA with an SMI system therefore offers a good solution for the real-time sensing.

In this chapter, a brief introduction to FPGA is given in Section 4.1. The overall measurement system based on FPGA is then described in Section 4.2. Section 4.3 presents the implementation of the FPGA design. The design is verified by using a
4.1 Introduction to FPGA

A field-programmable gate array is an integrated circuit designed to be configured by the customer or designer after manufacturing—hence “field-programmable”. FPGAs can be employed to accomplish any logical function that an application-specific integrated circuit (ASIC) could perform. They also have the ability to update their functionality after shipping, through a partial re-configuration of a portion of the design, which is advantageous for many applications. FPGAs contain programmable logic components, called “logic blocks”, and a hierarchy of reconfigurable interconnections that allow the blocks to be “wired together”—like many (changeable) logic gates that can be inter-wired in (many) different configurations. The most common FPGA architecture comprises an array of logic blocks (called Configurable Logic Block, CLB, or Logic Array Block, LAB, depending on the vendor), I/O pads, and routing channels. Generally, all the routing channels have the same width (number of wires).

Generally, the process of FPGA design includes module design, functional simulation, synthesize, implementation, timing verification, and board programming [62]. The flow chart is shown in Fig. 4-1.

- Module design

At this stage, designers describe the functions of their design to Electronic Design Automation (EDA) software. In the past, schematic diagrams were used for the input.
Nowadays, Hardware Description Language (HDL) is the most popular input method. The two used most widely types of HDL are Verilog HDL and very-high-speed integrated circuit HDL (VHDL). Both provide varying levels of abstraction to designers and allow top-down design, and are also convenient for the division and reuse of modules. The HDL used in this study is Verilog.

![FPGA design flow](image)

Figure 4-1: FPGA design flow.

- Function simulation

After the design is accomplished, special tools are needed to undertake function simulation to test whether the design meets the requirements. There are several common simulation tools, including Modelsim, Isim, etc. The simulation tool for this study use is Isim, provided by Xilinx.

- Synthesize
Synthesize stands for the process by which the HDL or schematic diagram design is translated into logic netlist consisting of basic logic elements, like and, or, flip-flop and so on. The logic netlist should obey the constraints and the objectives of the design.

- **Implementation**

  Implementation means the process by which the netlist from synthesize is to be implemented on an FPGA chip.

- **Board programming**

  A bitstream for devices is generated after an FPGA design is completely routed. This bitstream is used to configure FPGA chips.

### 4.2 System description

The overall system of SMI based $\alpha$ measurement is illustrated in Fig. 4-2. The sensing signal acquired from the SMI system is sent into the FPGA chip via an analog-to-digital converter (ADC). Pre-processing of the signal and measurement take place in the FPGA chip. The measurement results are then shown on the liquid crystal display (LCD). The de-noising method introduced in Chapter 2 is employed for the FPGA pre-processing design, and the measurement method that was described in [9], which will be retrieved in Section 4.3.3, is selected.
All components except the FPGA part of the system shown in Fig. 4-2 have already been introduced in Section 2.3. Here, only the FPGA board is introduced. The FPGA board used to implement this whole design is a Xilinx® Spartan-3E FPGA (XC3S500E-4FG320C) Development Board. The main features of this board are as follows [62]: 10478 logic elements (LEs); 128 Mbit Parallel Flash; 16 Mbits of SPI serial Flash; 20 multiplier and 232 maximum user I/O pins; a four-output SPI-based Digital to Analog Converter (DAC); a two-input SPI-based Analog-to-Digital Converter (ADC) with programmable-gain pre-amplifier; and an LCD display. The internal clock is generated by the 50-MHz oscillator on the development board.

4.3 System Implementation

4.3.1 A/D unit

The analog capture circuit consists of a programmable scaling pre-amplifier (Linear Tech
LTC6912-1 Dual Amp) and an ADC (LTC1407A-1 Dual A/D). From the datasheet [62], the range of the gain of the pre-amplifier is (-1 to 100) and its purpose is to scale the incoming voltage on VINA or VINB. The scaled input voltage signal is then sent into the ADC and the communication between them is based on a Serial Peripheral Interface (SPI) Bus. Afterward the ADC converts the analog voltage on VINA or VINB and converts it to a 14-bit digital representation and sends it into the FPGA chip via SPI.

The interfaces between this module and the FPGA chip are listed in Table 4-1.

<table>
<thead>
<tr>
<th>Interfaces</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clk</td>
<td>Input, clock from the board.</td>
</tr>
<tr>
<td>Rst</td>
<td>Input, reset signal, negative valid.</td>
</tr>
<tr>
<td>spi_miso</td>
<td>Input, FPGA ← AD, serial data: master input, serial output, which presents the digital representation of the sample analog values as two 14-bit two’s complement binary values.</td>
</tr>
<tr>
<td>simple_io</td>
<td>Output, which is used to test internal signal in the board.</td>
</tr>
<tr>
<td>amp_cs</td>
<td>Output, FPGA → Amp, active-low chip_select. The amplifier gain is set when the signal returns high.</td>
</tr>
<tr>
<td>amp_shdn</td>
<td>Output, FPGA → Amp, active-High shutdown, reset.</td>
</tr>
<tr>
<td>ad_conv</td>
<td>Output, FPGA → AD, active-High shutdown and reset.</td>
</tr>
<tr>
<td>spi_sck</td>
<td>Output, FPGA → Amp and AD, clock in SPI.</td>
</tr>
<tr>
<td>spi_mosi</td>
<td>Output, FPGA → Amp, serial data: master output, slave input, which presents 8-bit programmable gain settings.</td>
</tr>
<tr>
<td>platformflash_oe</td>
<td>Output, it is used to disable other devices on SPI bus to avoid contention.</td>
</tr>
<tr>
<td>strataflash_ce</td>
<td>The same use as platformflash_oe.</td>
</tr>
<tr>
<td>strataflash_we</td>
<td>The same use as platformflash_oe.</td>
</tr>
<tr>
<td>strataflash_oe</td>
<td>The same use as platformflash_oe.</td>
</tr>
</tbody>
</table>

The simulation results are shown in Fig. 4-3, from which it can be seen that the design meets the operating requirements of analog capture circuit according to [62].
4.3.2 Pre-processing unit

This section will describe the FPGA design of the pre-processing unit. The pre-processing unit contains three parts, a median filter, a wavelet filter and a normalization unit; its structure is as shown in Fig. 4-4. The median filter and wavelet filter have been described in Chapter 2. Each part of this design will be described in following part of this section.

Figure 4-3: Simulation results of the ADC module.

Figure 4-4: Basic structure of FPGA based SMI signal pre-processing.
4.3.2.1 Median filter FPGA based design

The median filter discussed in Chapter 2 is implemented in this section. As mentioned before, a five-point median filter is applied in our design. Before passing data into a median filter, a data acquisition block is needed to import five subsequent data samples, and four registers are used for this. The flow chart of a median filter is shown as below in Fig. 4-5. $R_{ij}$ represents registers and $C_{ij}$ denotes compare-swappers. The basic principle of this median filter is to sort the five samples and pick the median value to replace the current processed point. Taking the first level comparison for example, five data samples obtained from the acquisition block are sent into $C_{11}$ and $C_{12}$ in pairs and $R_{11}$. The compare-swappers compare their values and send the smaller output into the comparison block below at the second level, the larger one into the comparison block above, just as $C_{11}$ puts its smaller output into $C_{21}$ and the other into $C_{22}$. All comparisons at different levels operate similarly to this one. After five levels comparison, these data samples are descended and the median value is easily picked out. Furthermore, in order to accomplish high-speed computing, a special technique, pipeline design, is employed in this median filter. A pipeline is a set of data processing elements connected in series, so that the output of one element is the input of the next one, which makes the processing more efficient. In this median filter, five-level pipelines are employed since there are five-level comparisons. When one level comparison is completed, the output of this level comparison is sent as the input of the next level.
comparison, and the next set of data is sent into this level comparison. All the levels of comparison are processed with different input simultaneously. For example, when input samples $g[15]$-$g[19]$ are under comparison at the third level, the next five points $g[16]$-$g[20]$ do not wait for the comparison of $g[15]$-$g[19]$ to finish, but are already at the second level. The samples $g[17]$-$g[21]$, meanwhile, have reached the first level.

![Data acquisition block and Median filter diagram](image)

Figure 4-5: Flow chart of the design of the median filter; $g[k]$ is a data sample in SMI signals.

After the design is completed in Xilinx, the simulation results obtained are shown in the following figure. Ports “din” and “dout” are the input and output of this module, respectively. Other ports are internal signals. From ports “cur” to signal “pre4”, five subsequent sample points are shown. From this graph, it can be seen that the median value is picked out after five clock periods.
The wavelet transform based filter has been described in details in Chapter 2. This wavelet filter consists of four types of blocks: wavelet decomposition blocks, wavelet reconstruction blocks, a sharp transition detection block and down- and up-sampling blocks.

- **FIR filter**

The core of this wavelet filter consists of low- and high-pass filters. An output of an N-order FIR can be derived as linear convolution:

\[ y[n] = \sum_{k=0}^{N-1} H[k]g[n-k] \quad (4.1) \]

where \( g[n] \) is a data sample of input signal and \( H[k] \) are the coefficients of a FIR filter. The selected wavelet base is a Daubechies four-tap wavelet, and the corresponding coefficients are as follows: low-pass decomposition filter \{0.1294 0.2241 0.8365 0.4830\}; high-pass decomposition filter \{-0.4830 0.8365 -0.2241 -0.1294\}; low-pass reconstruction filter: \{0.4830 0.8365 0.2241 -0.1294\}; and high-pass reconstruction filter \{-0.1294 0.8365 0.2241 -0.4830\}.
-0.2241 0.8365 -0.4830. However, in an FPGA board, only integers can be stored and processed, so making all the coefficients integers is the first step. In hardware, numbers are represented in binary code. In order to make the coefficients integers, a multiplier $2^n$ is taken to enlarge them. Correspondingly, numbers multiplied by $2^n$ means the numbers are shifted left $n$ bits. In this project, $n=16$ is chosen. Then the Daubechies four-tap coefficients become: low-pass decomposition filter{-8480 14687 54821 31654}; high-pass decomposition filter{-31654 54821 -14687 -8480}; low-pass reconstruction filter{31654 54821 14687 -8480}; high-pass reconstruction filter: {-8480 -14687 54821 -31654}.

The equations of the FIR filters then become:

$$y[n] = -8480g[n] + 14687g[n-1] + 54821g[n-2] + 31654g[n-3] \quad (4.2)$$

$$y[n] = -31654g[n] + 54281g[n-1] - 14687g[n-2] - 8480g[n-3] \quad (4.3)$$

$$y[n] = 31654g[n] + 54821g[n-1] + 14687g[n-2] - 8480g[n-3] \quad (4.4)$$

$$y[n] = -8480g[n] - 14687g[n-1] + 54821g[n-2] - 31654g[n-3] \quad (4.5)$$

The illumination of the FIR is as shown in Fig. 4-7:

Figure 4-7: Design of FIR in four-tap Daubechies wavelet.
In this figure, \( R \) denotes registers and \( C1 \sim C4 \) stand for coefficients of FIR.

This module is simulated for verification and the results of the low-pass decomposition filter are shown in Fig. 4-8. Signals “product1” to “product4” are internal signals which are the products in Eq. (4.2). Signals “din” and “dout” are the input and output of this module, respectively. In order to confirm its accuracy, the same simulation is applied in Matlab with its integrated function for convolution. The results from Matlab are \{58755 151437 336801 429483 522165 614847 707529 893491 833096 316540\}. From the comparison of these two results, it can be seen that the FIR filter works correctly.

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>product1</td>
<td>10240</td>
</tr>
<tr>
<td>product2</td>
<td>17614</td>
</tr>
<tr>
<td>product3</td>
<td>603031</td>
</tr>
<tr>
<td>product4</td>
<td>166924</td>
</tr>
<tr>
<td>din</td>
<td>96575</td>
</tr>
<tr>
<td>dout</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>product1</td>
<td>10240</td>
</tr>
<tr>
<td>product2</td>
<td>17614</td>
</tr>
<tr>
<td>product3</td>
<td>603031</td>
</tr>
<tr>
<td>product4</td>
<td>166924</td>
</tr>
<tr>
<td>din</td>
<td>96575</td>
</tr>
<tr>
<td>dout</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 4-8: Simulation results of the FIR filter.

- **Sharp transition detection FPGA design**

A sharp detection block is the key to this filter. A differential unit is employed to detect the positions of sharp transitions in SMI signals. This detection block is simply realized by subtractions between two adjacent points in the SMI signals. A pulse train is obtained after this detection, where the pulses reflect the position of the sharp transitions in SMI signals. However, this pulse train is of varying magnitude. Thus, a threshold should be
pre-set. Considering the difference between the magnitudes of pulses and noises, half of the magnitude of the first pulse is set as a threshold and is stored in a register. The following magnitudes of each differential value should be compared with this threshold. Those above the threshold are seen as pulses and those below it are seen as noise. Combining this information with the reconstructed high frequency component, the signals are processed.

The module is synthesized in Xilinx ISE and the simulation results are shown in Fig. 4-9. Signal “sub_result” presents subtractions between two adjacent data samples and “select_out” is an indicating signal; when there is a sharp transition in the input signal, it turns to 1, and in other cases, 0. From the figure, it can be seen that when “sub_result” signal is very large or very small, ”select_out” signal turns to 1, which means there is a sharp transition in the input signal. Therefore, this sharp transition unit works well.

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>209,000 Hz</th>
<th>230,000 Hz</th>
</tr>
</thead>
<tbody>
<tr>
<td>select_out</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>sub_result</td>
<td>13</td>
<td>-5</td>
<td>-5</td>
</tr>
<tr>
<td>input</td>
<td>97</td>
<td>59</td>
<td>59</td>
</tr>
</tbody>
</table>

Figure 4-9: Simulation results of sharp transition detect module.

- **Down- and up- Sampling Unit**

After passing the low- and high-pass filters, down- and up-sampling are applied to all frequency contents to reconstruct the frequency component at different scales. This down-sampling operation is undertaken to discard odd terms, and up-sampling can be
completed by interpolating zeros between two adjacent terms in the down-sampled frequency content. Since an up-sampling is taken just after the down-sampling, these two operations can be accomplished by using one first-in-first-out (FIFO) Intellectual Property (IP) core provided by Xilinx. The design of this part is shown in Fig. 4-10. The meanings of ports and blocks in this design: clk, clock of this part; rst_n, reset signal; data_in, input of data; r_en and w_en are the reading and writing enable pin, respectively. After the data outputs from an FIFO block, a multiplexer (MUX) is employed to preserve the even terms and change the odd terms into zeros, thus accomplishing down- and up-sampling at the same time. The MUX is controlled by a counter, which only has two states, 0 and 1, to determine when to pass the terms and when to convert them to zeros.

![Figure 4-10: Design of down- and up-sampling using FIFO.](image)

This module has many interfaces. In order to present them conveniently, their definitions are listed in Table 4-2:

The simulation results are given in Fig. 4-11. It can be seen that when the “rst” signal is low and both the “rd_en” and “wr_en” are high, this unit starts to output the expected
series.

Table 4-2: Definitions of ports in the down_up module.

<table>
<thead>
<tr>
<th>Interfaces</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clk</td>
<td>Input, clock from the board.</td>
</tr>
<tr>
<td>Rst</td>
<td>Input, reset signal, negative valid.</td>
</tr>
<tr>
<td>rd_en</td>
<td>Input, read enable.</td>
</tr>
<tr>
<td>wr_en</td>
<td>Input, write enable.</td>
</tr>
<tr>
<td>din</td>
<td>Input, 14 bit two’s complement digital representation.</td>
</tr>
<tr>
<td>dout</td>
<td>Output, 14 bit two’s complement digital representation.</td>
</tr>
<tr>
<td>empty</td>
<td>Output, flag shows when the FIFO is empty.</td>
</tr>
<tr>
<td>full</td>
<td>Output, flag shows when the FIFO is full.</td>
</tr>
</tbody>
</table>

Figure 4-11: Simulation results of down- and up- sampling unit.

4.3.2.3 Normalization unit FPGA based design

Normalization is a crucial step in the pre-processing before measurement. In both of the method outlined in [9] and the method described in last chapter, zero-crossing points in the SMI signal waveform are used as characteristic points. Furthermore, in the model represented in Section 1.2.2, Eq. (1.27) indicates that an SMI signal should be in the range of [-1, 1] to do inverse cosine function. However, SMI signals obtained from the experimental set-up do not cover the range [-1, 1], which affects the position of
zero-points and the measurement of $\alpha$. So normalization of SMI signal is necessary. After filtering SMI signals, their normalization is no longer affected by sparkle-like noise. The key to normalization, therefore, is to remove the direct current component. This normalization can be derived from the following equation:

$$g_p[n] = ((g_f[n] - \min(g_f[n])) / (\max(g_f[n]) - \min(g_f[n])) - 0.5) \times 2 \quad (4.6)$$

In Eq. (4.6), $g_f[n]$ is a filtered SMI signal and $g_p[n]$ represents the normalized signal. $\min(g_f[n])$ and $\max(g_f[n])$ are the minimum and the maximum values of $g_p[n]$, respectively. However, FPGA implementation is too complicated to accomplish division in Eq. (4.6). So a compromise design is given, which can be described by Eq. (4.7):

$$g_p[n] = g_f[n] - (\max(g_f[n]) + \min(g_f[n])) / 2 \quad (4.7)$$

The center of the values of data samples in the signal is taken as zero. Therefore, the maximum value of the processed signal equals the absolute value of the minimum value of that. Maximum and minimum values can be found using a register. The first come-in value of a segment in $g_f[n]$ is stored as an extreme value, and the next value compares with the stored one. The next point replaces the stored extreme value if it is larger than the maximum value or smaller than the minimum value. Here, a part of the input signals needs to be stored for the next module. A random access memory (RAM) is employed for this. RAM is a form of computer data storage where data can be accessed randomly. When the write-enable is effective, the data is written into the RAM; meanwhile, the process of seeking the maximum and minimum values is working. When the writing process ends, the maximum and minimum values are obtained. When the read-enable is
effective, the RAM outputs the data processed with Eq. (4.7). The functions of each port are listed in Table 4-3:

Table 4-3: Definitions of ports in the normalization unit module.

<table>
<thead>
<tr>
<th>Interfaces</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>clk</td>
<td>Input, clock from the board.</td>
</tr>
<tr>
<td>en_normal</td>
<td>Input, enable signal of the module.</td>
</tr>
<tr>
<td>we</td>
<td>Input, read/write enable. When it’s 1, data is written into RAM. When it’s 0, data is read from RAM.</td>
</tr>
<tr>
<td>addr</td>
<td>Input address of the RAM input.</td>
</tr>
<tr>
<td>din</td>
<td>Input, 14 bit two’s complement digital representation.</td>
</tr>
<tr>
<td>dout</td>
<td>Output, 14 bit two’s complement digital representation.</td>
</tr>
</tbody>
</table>

Figure 4-12 describes the simulation results. Because of the limitation on space, only the first few data samples are shown in the following figure. The left sub-figure in Fig. 4-12 represents the first several input samples and the right gives the first few output data. The maximum value in the RAM is 989 and the minimum is -20. So the output {-504, -503, -502…} agrees with Eq. (4.7).
4.3.3 \( \alpha \) measurement unit design

4.3.3.1 Top level of measurement FPGA design

In order to accomplish an FPGA design for \( \alpha \) measurement, an appropriate method must first be chosen. There are two methods available: one is the method proposed in Chapter 3, and the other is the method in [9]. The method described in the previous chapter uses the method [60] to estimate \( C \) first, which requires phase unwrapping [21], which is too complicated to be implemented using FPGA. On the other hand, in [9] there are only four intervals that need to be measured, so this method is more appropriate to be selected. The core strategies of this method are summarized as following:

1) Measuring time intervals between characteristic points, as shown in Fig. 4-13.

![Figure 4-13: Illustration of the time intervals to be measured.](image)

2) Using the following equations to solve \( \alpha \) and \( C \).
\[
\phi_1 = \sqrt{C^2 - 1} + \frac{C}{\sqrt{1 + \alpha^2}} + \arccos\left(-\frac{1}{C}\right) - \arctan(\alpha) + \frac{\pi}{2} \tag{4.8}
\]

\[
\phi_2 = \sqrt{C^2 - 1} - \frac{C}{\sqrt{1 + \alpha^2}} + \arccos\left(-\frac{1}{C}\right) + \arctan(\alpha) - \frac{\pi}{2} \tag{4.9}
\]

The relationship between the phases and time intervals are given in [9]. Based on this method, an FPGA design for measurement of \( \alpha \) is proposed. This top level design is demonstrated by a state machine in Fig. 4-14.

![State Machine Diagram](image)

Figure 4-14: State machine of FPGA design on measurement of \( \alpha \).

In the top level, eight states (STAs) are divided to accomplish the measuring method. Descriptions of all the states are summarized in Table 4-4. The purpose of the following three states, STA_receive, STA_initial and STA_window, is to acquire a set of data whose length is a little longer than the length of one fringe of an SMI signal, and the initial point of which is an extreme point of the fringe. This fringe is used to measure \( \alpha \) and \( C \).

The next two states, STA_locating and STA_measuring, are used to measure the time intervals in the fringe. After two intervals in one fringe are measured, there is a branch.
In a state loop, if only one fringe is used, the next state will be STA_wait, and it will go to STA_receive to obtain the time intervals of the other fringe. After time intervals of both up and down fringes are obtained, $C$ and $\alpha$ are obtained in STA_Calpha.

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STA_idle</td>
<td>Initial state, reset.</td>
</tr>
<tr>
<td>STA_receive</td>
<td>Store a set of data whose length is about three fringes in RAM1 and find out the maximum value and minimum value in the first one-fringe-length data.</td>
</tr>
<tr>
<td>STA_initial</td>
<td>Determine whether the stored data are up or down fringes and select an extreme value as the initial point of the wanted fringe.</td>
</tr>
<tr>
<td>STA_window</td>
<td>Store a set of data whose length is about one fringe in RAM2 and find out the other extreme value.</td>
</tr>
<tr>
<td>STA_locating</td>
<td>Locate the addresses of the maximum point, the zero-crossing point and the minimum point in RAM2.</td>
</tr>
<tr>
<td>STA_measuring</td>
<td>Calculate time intervals needed.</td>
</tr>
<tr>
<td>STA_wait</td>
<td>Wait for half a period of an SMI signal.</td>
</tr>
<tr>
<td>STA_Calpha</td>
<td>Obtain the values of $C$ and $\alpha$.</td>
</tr>
</tbody>
</table>

4.3.3.2 STA_receive

This state is designed to store data whose length is about three fringes and find out the maximum and minimum values in the first one-fringe-length data. Based on investigation, one fringe includes about 100 data samples. The storage function is realized by using a RAM, called RAM1 to distinguish it from the RAM used in the following sections. When the write enable of RAM1 is effective, data is written into RAM1. When read enable is
effective, data is output for the utilization of other modules. The width of RAM1 is 14 bit, which is the same as the input data and the depth is pre-set as 300. Simultaneously, two registers are used to find out the extreme values of these data where the same method is used to seek extreme values in the normalization unit. Here a counter is used to control the process of seeking extreme values. When the counter is less than 100, the extreme-value seeking process is running. Otherwise, it stops. This operation is performed to make the initial point in the first 100 data samples and to make sure there is at least a whole fringe left.

The module is synthesized in Xilinx, and the simulation results of this module are given in Fig. 4-15. Signal “we” is the read/write enable. When it is 1, data is written into the RAM. When it is 0, data is read from the RAM. Signal “cs_n” is the chip select signal, which is negative valid and “address” is the data address in the RAM. Only part of the results can be shown due to limited space. From the figure, the maximum of the first 100 value of the input of this module is 59, which is as shown, and the minimum is also given correctly. Although there is higher value in the left data in RAM1, the maximum and minimum value remains unchanged, which agrees with the purpose of design.
4.3.3.3 STA_initial

The main purpose of STA_initial is to find the initial point for the intended stored fringe. It also produces a judgment on whether the stored data is in the up or down fringe of the SMI signal. In order to accomplish this, the module designed in this state first calculates the sum of the maximum and minimum values of the data in RAM1. Since the data in RAM1 are normalized, if they are in the up fringe, the absolute value of the maximum value is bigger than that of the minimum, and thus, the sum of them is a positive number. Otherwise, it is a negative number. The judgment is made on this basis. If data in RAM1 are in the up fringes, the minimum point is selected as the initial point of a fringe. Otherwise, the maximum point is selected. This operation is via comparison between the stored value and the known extreme value while RAM1’s read enable is effective. When the output data point equals the extreme value, this state is ended and the process turns into the STA_window.

The module is synthesized in Xilinx and definitions of its interfaces are given as follows.

<table>
<thead>
<tr>
<th>Interfaces</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clk</td>
<td>Input, clock from the board.</td>
</tr>
<tr>
<td>en_initial</td>
<td>Input, enable signal.</td>
</tr>
<tr>
<td>Max</td>
<td>Input, maximum value of the first 100 data samples in RAM1.</td>
</tr>
<tr>
<td>min</td>
<td>Input, minimum value of the first 100 data samples in RAM1.</td>
</tr>
<tr>
<td>initial_flag</td>
<td>Output, a flag signal. When it turns to 1, this state is ended.</td>
</tr>
</tbody>
</table>
The simulation results are given in Fig. 4-16. It can be seen that the judgment as to whether the data is in the up or down fringe is correct. When the input equals to the maximum value, the initial_flag turns to 1, which means that this state is ended.

![Simulation results of the initial module.](image)

### 4.3.3.4 STA_window

The objective of this state is to store one-fringe-length data into another RAM, RAM2. The other extreme values of this set of data also need to be searched since the initial point is an extreme value of this data. From the initial point, RAM1 transfers its data into RAM2. The initial point is already set. When the data is being transferred, the seeking process is operating. If the initial point is a minimum value, then the maximum value in RAM2 needs to be found. Otherwise, the minimum value must be searched. The same method of seeking extreme values with the normalization unit using a register is applied here. The data in RAM2 may not be exactly a whole fringe. In most cases, it must be a little longer than one fringe. However, this does not affect the measurement. According
to the waveform in Fig 4-13, there is a sharp transition between two fringes, which
means an extreme point is followed by a reverse one. The longer part in RAM2 is similar
to the initial part of the stored fringe, which cannot influence the process of seeking the
extreme point that is the reverse of the initial point. The module is synthesized in Xilinx
ISE, and definitions of each port are listed as below.

Table 4-6: Definitions of ports in the window module.

<table>
<thead>
<tr>
<th>Interfaces</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clk</td>
<td>Input, clock from the board.</td>
</tr>
<tr>
<td>en_initial</td>
<td>Input, module enable signal, positive valid.</td>
</tr>
<tr>
<td>up_down</td>
<td>Input, 0 stands for that it’s an up fringe and 1 tells that it’s a down fringe.</td>
</tr>
<tr>
<td>addr</td>
<td>Input, address of RAM2.</td>
</tr>
<tr>
<td>we_window</td>
<td>Input, write/read enable of RAM2, 0 gives reading operation and 1 indicates writing.</td>
</tr>
<tr>
<td>din</td>
<td>Input, 14 bit two’s complement digital representation read from RAM1.</td>
</tr>
<tr>
<td>dout</td>
<td>Output, 14 bit two’s complement digital representation.</td>
</tr>
<tr>
<td>max_window</td>
<td>Input, maximum value of the first 100 data samples in RAM1.</td>
</tr>
<tr>
<td>min_window</td>
<td>Input, minimum value of the first 100 data samples in RAM1.</td>
</tr>
</tbody>
</table>

The simulation results are shown in Fig. 4-17. From the figure, it can be seen that when
signal “we_enable” changes from 1 to 0, the signal “dout” starts to output data stored in
RAM2 while signal “we_enable” is 1. Another notification is that when signal “up_down”
is 1, the minimum value is given in signal “min_window”. The design of this module
meets the requirements.
4.3.3.5 STA_locating

The main purpose of STA_locating is to determine the locations of characteristic points, extreme points and zero-crossing points. The method used to find out the positions of extreme points is very simple by acquiring their addresses in RAM2. Let the read enable of RAM2 be effective and the data is read. Simultaneously, the data is compared with the extreme value obtained in the STA_window. When the extreme value equals the data read from RAM2, the address of this output point is recorded as the position of this extreme value. Determining the zero-crossing point is a little more complicated. As the data is represented as 14 bit two’s complement code, the sign-bits of negative numbers are 1 and the sign-bits of positive numbers are 0. Therefore, when the sign-bits of two successive points are different, these two points are zero-crossing points and their addresses are recorded as their locations. Since the determination is based on the change of sign-bit, there is a risk that sharp transitions may be confused with zero-points. In order to distinguish them, a threshold of the difference of two successive points has to
be pre-set. When the difference is less than the threshold, the point is a zero-crossing point. Otherwise, it is a sharp transition. The maximum values can be set as the threshold, because the difference between the maximum value and the minimum value is greater than the maximum value.

Table 4-7: Definitions of ports in the locating module.

<table>
<thead>
<tr>
<th>Interfaces</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clk</td>
<td>Input, clock from the board.</td>
</tr>
<tr>
<td>en_initial</td>
<td>Input, module enable signal, positive valid.</td>
</tr>
<tr>
<td>up_down</td>
<td>Input, 0 stands for that it’s an up fringe and 1 tells that it’s a down fringe.</td>
</tr>
<tr>
<td>addr</td>
<td>Input, address of RAM2.</td>
</tr>
<tr>
<td>we_window</td>
<td>Input, write/read enable of RAM2, 0 gives reading operation and 1 indicates writing.</td>
</tr>
<tr>
<td>din</td>
<td>Input, 14 bit two’s complement digital representation read from RAM1.</td>
</tr>
<tr>
<td>dout</td>
<td>Output, 14 bit two’s complement digital representation.</td>
</tr>
<tr>
<td>max_window</td>
<td>Output, maximum value of an up fringe in RAM2.</td>
</tr>
<tr>
<td>min_window</td>
<td>Output, minimum value of a down fringe in RAM2.</td>
</tr>
</tbody>
</table>

The simulation results are given in Fig. 4-18. The input we set is an up fringe. Therefore, the positions we need to find out are the maximum point and the zero-crossing point. The design gives them correctly.
Figure 4-18: Simulation results of the locating module.

4.3.3.6 STA_measuring

This is a simple module which calculates $t_{13}$, $T_1$, $t_{24}$ and $T_2$ using position information given in STA_locating. In this module, the locations obtained in last state are used to calculate them using Equations (4.10)-(4.13).

\[
\begin{align*}
  t_{13} &= \text{max}_\text{up}_\text{location}-\text{zero}_\text{up}_\text{location} \quad (4.10) \\
  T_2 &= \text{max}_\text{up}_\text{location} - 1 \quad (4.11) \\
  t_{24} &= \text{min}_\text{down}_\text{location}-\text{zero}_\text{down}_\text{location} \quad (4.12) \\
  T_2 &= \text{min}_\text{down}_\text{location} - 1 \quad (4.13)
\end{align*}
\]

The module is synthesized in Xilinx ISE, and each of its port shown in Table 4-8. The four location signals have the same meanings within Table 4-7.

<table>
<thead>
<tr>
<th>Interfaces</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clk</td>
<td>Input, clock from the board.</td>
</tr>
<tr>
<td>en_measure</td>
<td>Input, module enable signal, positive valid.</td>
</tr>
</tbody>
</table>
up_down Input, 0 stands for an up fringe and 1 for a down fringe.
up_t13 Output, time interval t_13.
up_T1 Output, time interval T_1.
down_t24 Output, time interval t_24.
down_T2 Output, time interval T_2.

The simulation results are shown in Fig. 4-19. Obviously, this module can give the correct results.

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>up_t13[4:0]</td>
<td>67</td>
</tr>
<tr>
<td>up_t21[5:0]</td>
<td>96</td>
</tr>
<tr>
<td>down_t20[6:0]</td>
<td>40</td>
</tr>
<tr>
<td>down_T20[6:0]</td>
<td>97</td>
</tr>
<tr>
<td>measure_over</td>
<td>1</td>
</tr>
<tr>
<td>max_up_location_15</td>
<td>97</td>
</tr>
<tr>
<td>min_down_location</td>
<td>88</td>
</tr>
<tr>
<td>zero_up_location</td>
<td>30</td>
</tr>
<tr>
<td>zero_down_location</td>
<td>28</td>
</tr>
<tr>
<td>up_down</td>
<td>1</td>
</tr>
<tr>
<td>clk</td>
<td>1</td>
</tr>
<tr>
<td>en_measure</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 4-19: Simulation results of the measuring module.

4.3.3.7 STA_CalphaLUT

From last state, four time intervals have been already acquired. Thus, the last step is to solve Equations (4.8)-(4.9) to get \( C \) and \( \alpha \). Obviously, Equations (4.8)-(4.9) are too complicated to be solved in FPGA. In order to avoid complex computing, look-up table (LUT) method is chosen to give the values of \( C \) and \( \alpha \). A look up table is a memory with a one-bit output that essentially implements a truth table where each input combination generates a certain logic output. Firstly, \( t_{13} / T_1 \) and \( t_{24} / T_2 \) are calculated using a divider Intellectual Property (IP) core provided by Xilinx. Secondly, Equations (4.9)-(4.10) are solved in Matlab. Finally, the corresponding relationship between \( t_{13} / T_1 \),...
$t_{24}/T_2$, $C$ and $\alpha$ are written into LUTs. Based on the input, $C$ and $\alpha$ can now be acquired. The module and its ports are given below:

<table>
<thead>
<tr>
<th>Interfaces</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>clk</td>
<td>Input, clock from the board.</td>
</tr>
<tr>
<td>en_Calpha</td>
<td>Input, module enable signal, positive valid.</td>
</tr>
<tr>
<td>up_t13</td>
<td>Input, time interval $t_{13}$.</td>
</tr>
<tr>
<td>up_T1</td>
<td>Input, time interval $T_1$.</td>
</tr>
<tr>
<td>down_t24</td>
<td>Input, time interval $t_{24}$.</td>
</tr>
<tr>
<td>down_T2</td>
<td>Input, time interval $T_2$.</td>
</tr>
<tr>
<td>Error</td>
<td>Output, if the feedback level is out of the range of measurement, error will turn to 1.</td>
</tr>
<tr>
<td>C</td>
<td>Output, feedback level factor.</td>
</tr>
<tr>
<td>alpha</td>
<td>Output, linewidth enhancement factor.</td>
</tr>
<tr>
<td>measuring_over</td>
<td>Output, a flag. When it turns from 0 to 1, this module is finished.</td>
</tr>
</tbody>
</table>

The simulation results of this module are demonstrated in Fig. 4-20. In this figure, it can be seen that $C$ and $\alpha$ are given in a decimal format as a fractional format cannot be displayed. For example, $C = 18$ actually stands for $C = 1.8$ and $a = 34$ is $a = 3.4$.

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>1, 000,000 ps</th>
<th>2, 000,000 ps</th>
<th>3, 000,000 ps</th>
</tr>
</thead>
<tbody>
<tr>
<td>$[5:0]$</td>
<td>18</td>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$[8:0]$</td>
<td>24</td>
<td></td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>$[8:0]$</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>en_Calpha</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>clk</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$[8:0]$</td>
<td>70</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>up_t13</td>
<td>98</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>up_T1</td>
<td>98</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>down_t24</td>
<td>40</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>down_T2</td>
<td>88</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 4-20: Simulation results of the CalphaLUT module.
4.3.3.8 Simulation of all modules together

After the design of all the modules is completed, interfaces between them, such as enable signals, chip select signals and write/read enables, are employed to manage their relationships. The simulation results of the combination of all the modules are illustrated in Fig. 4-21, which shows that \( C \) and \( \alpha \) can be measured and that the system works well.

![Figure 4-21: Simulation results of the measurement method.](image)

4.3.4 System display

The last step of this FPGA based measurement design is to show the results of \( C \) and \( \alpha \) on the liquid crystal display (LCD). The LCD on the Spartan 3e Starter Kit board has a Character Generator ROM (CG ROM) which contains the font bitmap for each of the predefined characters that it can display, as shown in Fig. 4-22 [62]. Taking digital “7” as an example, the upper data nibble “0011” are sent into the interface between FPGA chip and LCD, then the upper data nibble “0111”. These data nibbles correspond to the rows and lines in Fig. 4-22. Thus the first thing in this module is to represent the value of \( C \) and \( \alpha \) by two four-bit nibbles, one of which is for integer digits and the other for
fractional digits. For instance, if the input $C = 27$ from last module, the first step is to make sure which interval it belongs to. Integers are divided into different intervals, like [20, 30) and [30, 40). Thus the upper nibble of the integer digit is “0011” and the lower nibble is “0010”. Then the fractional digit is computed by using 20 to subtract 27, which makes the upper nibble of the integer digit “0011” and the lower nibble “0111”.

![Image](image_url)

Figure 4-22: LCD character set.

The definitions of the ports of the module are listed in Table 4-10.

<table>
<thead>
<tr>
<th>Interfaces</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clk</td>
<td>Input, clock from the board.</td>
</tr>
<tr>
<td>en_Calpha</td>
<td>Input, module enable signal, positive valid.</td>
</tr>
<tr>
<td>up_t13</td>
<td>Input, time interval $t_{13}$.</td>
</tr>
<tr>
<td>up_T1</td>
<td>Input, time interval $T_{1}$.</td>
</tr>
<tr>
<td>down_t24</td>
<td>Input, time interval $t_{24}$.</td>
</tr>
<tr>
<td>down_T2</td>
<td>Input, time interval $T_{2}$.</td>
</tr>
<tr>
<td>error</td>
<td>Output, if the feedback level is out of the range of measurement, error will turn to 1.</td>
</tr>
<tr>
<td>C</td>
<td>Output, feedback level factor.</td>
</tr>
<tr>
<td>alpha</td>
<td>Output, linewidth enhancement factor.</td>
</tr>
<tr>
<td>measuring_over</td>
<td>Output, a flag. When it turns from 0 to 1, this module is finished.</td>
</tr>
</tbody>
</table>
The simulation results are shown in Fig. 4-23. It can be seen that the inputs $C$ and $\alpha$ are transferred to the format that the LCD can read.

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ci_low[3:0]</td>
<td>0010</td>
</tr>
<tr>
<td>C2_low[3:0]</td>
<td>1001</td>
</tr>
<tr>
<td>alpha1_low[3:]</td>
<td>0101</td>
</tr>
<tr>
<td>alpha2_low[3:]</td>
<td>0010</td>
</tr>
<tr>
<td>transition_ov</td>
<td>1</td>
</tr>
<tr>
<td>C[5:0]</td>
<td>29</td>
</tr>
<tr>
<td>alpha[5:0]</td>
<td>52</td>
</tr>
<tr>
<td>clk</td>
<td>0</td>
</tr>
<tr>
<td>en_transition</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 4-23: Simulation results of the LCD module.

### 4.4 System test

All FPGA designs are combined and transferred to a bit-stream file by Xilinx ISE. Then the bit-stream file is downloaded into the FPGA Spartan 3e development board. The FPGA based SMI system is completed and the values of $C$ and $\alpha$ can be read from the LCD. The photograph of the experimental is shown in Fig. 4-24, in which each component of the system is labeled. The black box is the controlling circuit, and other components, LD, PD and FPGA board have been already introduced.
When the system is working, an SMI signal can be detected from the PD. It is shown on the oscilloscope, the photo of which is shown in Fig. 4-25. This signal can be sent into a PC and processed in Matlab. Using the method described in Chapter 2 to de-noise it and the approach in Chapter 3 to measure, the values of $C$ and $\alpha$ are $C=1.9$ and $\alpha=3.6$

After the PD is connected to FPGA board, and the FPGA board is configured with the design, the SMI signal is sent into the board. The values of $C$ and $\alpha$ can be read from the LCD. It is as shown in Fig. 4-26. The results are $C=1.9$ and $\alpha=3.6$, which agrees with the values measured in a PC. The performance of this FPGA design is reliable, so
this design meets the requirements.

Figure 4-26: Measurement results on the LCD.

4.5 Summary

In this chapter, an FPGA design of an SMI based $\alpha$ measurement method is implemented. The SMI signals are captured by the analog capture circuit controlled by FPGA chips and represented as a 14-bit two’s complementary code. Then a five-point median filter is accomplished to deal with the sparkle-like noise. After that, FPGA design of the wavelet transform filter is implemented, which consists of four different units: FIR filters, down- and up-sampling blocks and a sharp transition detection unit. As a part of pre-processing, a simple normalization block is also implemented. After pre-processing, the SMI signal is adopted to measure $\alpha$ and $C$ using the method in [9]. At last, the measurement values are shown on the LCD and compared with the results obtained from MATLAB, off-line. The results show good agreement.
Chapter 5  Conclusion

Self-Mixing Interferometry (SMI) sensing technique has been researched extensively in the field of instrument and measurement. In this thesis, a measurement method of linewidth enhancement ($\alpha$) that can work with a wide range of feedback level ($C$) is proposed. Since this method requires that the information on sharp transitions in the waveform to be precise, a de-noising method based on a wavelet transform filter and a median filter has designed to deal with practical SMI signals. Furthermore, in order to make the system a compact one, an FPGA design of the preprocessing method is implemented. The contributions of this thesis are summarized in Section 5.1.

5.1  Research contributions

- A pre-processing method for SMI signals is developed that can efficiently reduce noise in them while preserving sharp transitions in the waveform. The method is combined by a wavelet filter and a median filter. The wavelet filter is employed to eliminate high-frequency noise and the median filter is used to reduce sparkle-like noise.
A novel measuring method for $\alpha$ is achieved, which can expand the measurement range of feedback level. The method is based on SMI technique, which can broaden the measurement range of $C$ from about 3.2 to about 6.

FPGA design of $\alpha$ measurement is implemented, which can make the whole SMI system compact, and allow real-time measurement. The algorithm used in FPGA design includes two part, pre-processing and measurement, which employ the method described in Chapter 2 and method in [9].

5.2 Suggested future work

In further work on the pre-processing method for SMI signals and $\alpha$ measurement, the following topics are expected to be improved:

- The FPGA board used in this project is in low version. For example, only 14-bit digital two's complementary code is used to represent the input SMI signals. Advanced FPGA boards, like Spartan 6A and Vertex 6, could be employed for more precise calculation and higher speed.
- Rules of optimal selection of the fringes to do measurement is not determined because the time intervals of each fringe in one period are not the same, so which fringe is chosen to calculate $\alpha$ therefore remains an issue to be explored.
- This method described in the thesis can expand the measuring range up to $C=6$. But for the feedback with higher $C$ value, it is still a problem.
Appendix A: Reference


interference inside a single-mode diode laser for optical sensing applications,"


[57] D. Kuksenkov, S. Feld, C. Wilmsen, H. Temkin, S. Swirhun, and R. Leibenguth, "Linewidth and alphasfactor in AlGaAs/GaAs vertical cavity surface emitting


