Master's Thesis

Real-time Implementation Of Digital Cavity

Siyuan Fu
Aohan Jin

Department of Electrical and Information Technology,
Faculty of Engineering, LTH, Lund University, October 2014.

Real-time Implementation Of Digital Cavity

Siyuan Fu (soc11fsi@student.lu.se)
Aohan Jin (soc11aji@student.lu.se)

October 12, 2014

Abstract

This master's thesis work demonstrates a new methodology for high precision measurement of ultra-high frequency signals. The method is based on the theory of digital cavities that recently has been introduced [1]. It uses high sampling rate ADCs to digitize the analog signals to be measured, and then applies digital cavity algorithms on the digitized data. The Q-factor of a digital cavity can reach up to 10^9, which is significantly higher than that of alternatives on market.

Two experiments have been designed to verify and test the performance of the algorithm of the digital cavity. One more experiment has been designed to recover a signal from a very noisy background. The tests show the possibilities of future applications of digital cavities in extreme precision measurements of signals as well as other physical quantities.

In order to conduct real-time analysis of gigahertz signals, an FPGA-based digital cavity accelerator has been designed and implemented using the Xilinx XUPV5-LX110T platform. With the processing capability of the FPGA, the digital cavity can analyse signals of several gigahertz frequency at a Q-factor of up to 10^8 in real time.

The digital cavity algorithm has greatly increased the precision of ultra-high frequency signal measurements, which makes it promising for commercialization. Some commercialization schemes of the digital cavity have been discussed, including advantages and disadvantages to suit the different applications.

Contents

Chapter 1

Introduction

Precision measurement of signals plays a very important role in many fields of technology, such as radar applications and clock synchronization.

For example, radars measure the speed of a moving object by aiming a radio wave of a specified frequency towards the object, and monitoring the reflected beam. By analysing the difference between the outgoing wave and the reflected wave, the speed of the object can be calculated based on the Doppler effect. The more precisely one measures the frequency of the beam, the more precisely can the speed of the target can be measured.

As technology makes unceasing progress, the requirement precision measurements and analysis of signals are also becoming higher. Modern radars use microwave signals to gain the best resolution. Similarly, for the synchronization of clocks in some computing clusters, the difference in the clock rates have to be reduced to a level of $10^{-12}s$ per second. In these cases it would be advantageous to be able to measure the ultra-high frequency signals with extreme precision. The precision requirement usually means huge quantities of data within a short time interval, which limits the real-time processing of the signal.

In this master thesis, we have implemented a new method, which is known as digital cavity, to measure the frequency, phase and amplitude precisely. The basic theory of the digital cavity algorithm is presented in detail and its application as a comb filter in the generalized lock-in amplifier (LIA) are discussed. Three experiments have been designed to verify the theory, its performance, and possible applications. The first experiment is designed to test the minimum frequency resolution that can be achieved in a software based system. The tests are done by generating a 245 MHz sine signal by a signal generator that is sampled by a digitizer. A general-

ized LIA is applied to the digitized data to analyse its frequency components. The second experiment focuses on the algorithm's performance on measuring ultra-high frequency signals at 10 GHz. A promising application of digital cavity algorithm is measuring minor deformation of cables caused by temperature change which is also included in this experiment. The third experiment is designed to test the ability of a digital cavity in recovering signals from extremely noisy backgrounds.

All measurements mentioned above have been conducted using a software platform in a personal computer(PC). In practice, the computational speed of the PC limits the application of the digital cavity in higher frequency measurement. Therefore, a hardware accelerator has been implemented. The implementation is based on the Xilinx XUPV5LX110t platform, and consists of a digital cavity core, a reference signal generator, and a controller. With the hardware accelerator, the performance has been significantly improved and the calculation time has been shortened more than 10 times with the same precision. Thus, this implementation can be used in real-time analysis.

The digital cavity algorithm can achieve a Q-factor of up to 10^9, which is about 10^3 times higher than that of products on the market now. This makes the product using it very competitive and opens up for the prospects of commercialization. In the last part of this thesis, several schemes of commercialization are discussed, including their advantages and applications where they fit in.

Chapter 2

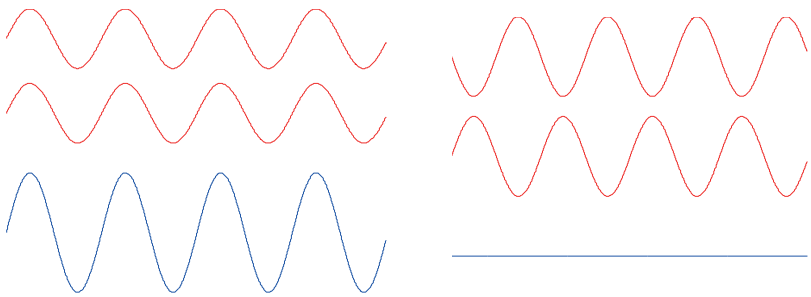
Background

Frequency and phase analysis of signals is widely used in radar applications, physics research, electronics, etc. Common methods of analysis can be divided into three kinds: frequency measurement using an analog cavity and analysis using a lock-in amplifier. Frequency measurements using an analog cavity is a traditional way of measuring the frequency of a signal. However, its performance is limited by the environment and the energy loss of the signal due to the bouncing between the ends of the analog cavity. The lock-in amplifier can have very narrow linewidth and it is routinely used to separate real signal from very noisy background. These approaches will be discussed in the following sections.

2.1 Analog cavity

A traditional way of measuring the frequency of a signal is to use a frequency meter, which is based on one or two analog cavities. The principle of such device relies on self-interference [2]. In the analog cavity of a frequency meter, the signal bounces between the two ends of the cavity, and interferes with itself. Since the reflection does not change the frequency of the signal, its frequency remains the same. For two signals with the same frequency, the interference can be either constructive or destructive, as shown in Figure 2.1.

In frequency meters, destructive interference is used to detect the signal. The length of cavity inside can be manually tuned. One end of frequency meter is connected with the signal being measured, the other end is connected to a detector and an indicator. The detector and the indicator are used to detect the power 'dip' caused by power absorbed by the cavity. The detector and indicator can be a



(a) Constructive (b) Destructive.

Figure 2.1: Interference.

thermistor and a power meter.

The frequency meter is one of the earliest equipments for signal measurement. Its performance is limited by many factors, such as the temperature and the observation error by operators. These flaws make it inappropriate for modern signal measurement.

2.2 Lock-in Amplifier

In the cases when the signal is very weak it is hard to distinguish it from the noise. Simple amplifiers can not be applied in such situations because they affect the noise too. In some cases the signal-noise ratio(SNR) deteriorates after the amplification. Lock-in amplifiers (LIA) are designed to detect and measure such small AC signals [3].

A lock-in amplifier (also known as a phase-sensitive detector) is a type of amplifier that can extract a signal with a known carrier wave from an extremely noisy environment. The device is often used to measure phase shift, even when the signals are large and of high signal-to-noise ratio, and do not need further improvement. The basic principle of a lock-in amplifier is the orthogonality of sinusoidal functions [4].

The most important component in a LIA is the phase-sensitive detector(PSD). The PSD can detect the signal with an extremely narrow bandwidth. With the gain remaining the same, narrow bandwidth means the reduction of noise, because the power of noise is proportional to the bandwidth.

The PSD requires a reference frequency. A phase-locked loop (PLL) is used to generate a phase tunable internal reference frequency from the external reference.

The output of the PSD is given by Equation (2.1) [5].

$$V_{psd}(t) = V_{sig}sin(\omega_{sig}t + \theta_{sig}) \cdot V_L sin(\omega_L t + \theta_L), \tag{2.1}$$

where $V_{psd}(t)$ is the output of the PSD, V_{sig} is the amplitude of the signal, $sin(\omega_{sig}t + \theta_{sig})$ presents frequency and phase of the signal, V_L is amplitude of the reference, and $sin(\omega_L t + \theta_L)$ with frequency ω_{sig} while V_L, ω_L and θ_L are the same for the reference. Transformation of Equation (2.1) gives

$$V_{psd}(t) = \frac{1}{2}V_{sig}V_L cos[(\omega_{sig} - \omega_L)t + \theta_{sig} - \theta_L] - \frac{1}{2}V_{sig}V_L cos[(\omega_{sig} + \omega_L)t + \theta_{sig} + \theta_L]. \tag{2.2}$$

Equation (2.2) shows that the output of the PSD is actually a combination of two AC signals. One of them is at the sum frequency of the signal and internal reference, and the other is at the difference.

$V_{psd}(t)$ passes through a narrow bandwidth low-pass filter. For general ω_L and ω_{sig}, almost nothing will be left after the low-pass filter. But if ω_L equals ω_{sig}, which means the frequency of the internal reference equals to that of the signal, the output after the filter is given by:

$$V_{psd} = \frac{1}{2} \cdot V_{sig} \cdot V_L cos(\theta_{sig} - \theta_L). \tag{2.3}$$

Since V_L is a constant, Equation (2.3) shows that when the frequencies and the phases of the reference and the signal are equal, the output of PSD is a DC value proportional to the amplitude of the signal.

Furthermore, an LIA can also be used in measuring the amplitude and phase of signals. In such cases, two PSD should be applied to the signal at the same time. The phase of one of the references should be offset by 90 degrees. Suppose the output of PSD1 is as shown in Equation (2.3), then the output of PSD2 is given by:

$$V_{psd2} = \frac{1}{2} \cdot V_{sig} \cdot V_L cos(\theta_{sig} - \theta_L - 90^\circ), \tag{2.4}$$

which is equal to

$$V_{psd2} = \frac{1}{2} \cdot V_{sig} \cdot V_L sin(\theta_{sig} - \theta_L). \tag{2.5}$$

Using Equation (2.3) and (2.5), the phase of the signal is given by:

$$\theta_{\text{sig}} = \arctan \frac{V_{\text{psd2}}}{V_{\text{psd1}}} + \theta_L \tag{2.6}$$

2.3 Conclusion

The lock-in amplifier based on analog cavity is a traditional method to detect signals among background noise, but the precision of such lock-in amplifiers is limited by the energy loss during the signal bouncing within the analog cavity. At the same time, it is also easily affected by the environment factors such as temperature and humidity. Digitization of the cavity can be a solution of these problems, which can be referred as digital cavity.

Chapter 3

Theory

A frequency meter using an analog cavity can be applied with a wide bandwidth, but will be limited by many factors, such as the influence of the environment and the observation error of operator. As theses devices only have a Q-factor of about 10^6 they cannot be used to analyze signals with very high precision.

To solve this problem, a digital cavity can be used.

3.1 Digital cavity theory

A digital and an analog cavity share the same basic principles. They both measure the signal using the principle of interference. The difference is that the analog cavity uses the interference of the analog signal directly, while the digital cavity uses an analog-to-digital converter (ADC) to transform the analog signal into digital data, and then uses the digital data to analyse the signal.

The response of a digital cavity is described by the following equation:

$$y(j \cdot \Delta t; n) := \sum_{k=0}^{N_c} x(j \cdot \Delta t + k \cdot T); j = 1, 2, \cdots n, \tag{3.1}$$

where y is the response of the cavity to the signal x, Δt is the digitization interval, N_c will be referred as fold number, which defines the amount of waveforms to be taken into the digital cavity [1].

According to the Equation (3.1), the digitized signal x is sectioned to N_c waveforms of n elements with an interval Δt of each element, and then summed up to generate a response. The length of the waveform, $n \cdot \Delta t$ is defined as the *cavity length*. This equation mimics the interference effect of the sinusoidal signals.

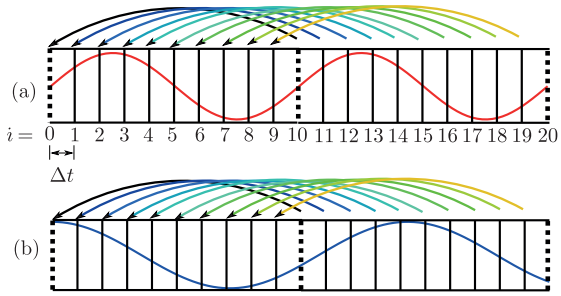


Figure 3.1: Digital cavity theory diagram.

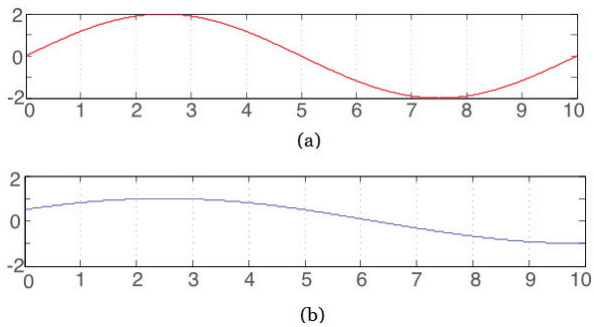


Figure 3.2: Example response of digital cavity.

Figure 3.1 shows the concepts of the digital cavity, where Figure 3.1(a) shows a signal resonant with the cavity while (b) shows a non-resonant signal. As described in the Figure 2.1, only those signals whose periods match the cavity length will resonate with the cavity, and they interfere constructively, while the others interfere destructively. Figure 3.2 is the example response of digital cavity for the signals shown in Figure 3.1. The red line corresponds to the signal shown in Figure 3.1(a) and blue line corresponds to 3.1(b). In Figure 3.1, there are 2 waveforms for each signal, so the N_c is 2. Figure 3.2 shows that with an N_c equals 2, the signal with resonant frequency gains more energy than the signals with non-resonant frequency. With a N_c large enough, the resonant signal will have a much larger amplitude compared to the others. This effect can suppress the noise with large frequency difference to the resonant frequency, and strengthen the signal, which makes it

possible for the digital cavity to be used as a lock-in amplifier.

For a digital cavity, another important factor is the linewidth [5]. The linewidth indicates the width of the main peak at the 1/2 amplitude. The linewidth of a digital cavity is given by:

$$2\xi_{f_0} \leqslant \frac{2\sqrt{2}f_0}{N_c + 1}, \tag{3.2}$$

where $2\xi_{f_0}$ is the full-width at half maximum of the response of the cavity and f_0 is the resonance frequency of the cavity. This frequency is given by:

$$f_0 = 1/(n \cdot \Delta t). \tag{3.3}$$

Equation (3.3) shows the possibility of tuning the resonance frequency by changing the number of contained elements of a waveform section n, or the time interval between elements Δt. For example, if a signal is digitized at 100 MS/s (mega samples per second), and n is chosen to be 100, then $f_0 = 1$ MHz. If $n = 90$ then $f_0 = 1.1111\cdots$ MHz. In this case $\Delta t = 0.01\mu s$. By providing the ADC with a tunable external clock, Δt can be tuned precisely. However, this method requires a high precision voltage controlled oscillator.

Another method to measure the frequency of the signal is to use a similar approach as in the LIA where the digital cavity forms a part of the PSD scheme. The difference is that only one reference is needed, thus no quadrature reference is necessary. As described in Equation (3.4),

$$V_{\text{Input}} = \frac{A}{2}\{sin[(\omega_{\text{sig}} - \omega_{\text{shift}})t + \phi] + sin[(\omega_{\text{sig}} + \omega_{\text{shift}})t + \phi]\}. \tag{3.4}$$

the multiplication of signal and reference gives the combination of two components wherein one is the sum frequency and the other has the difference . Here the reference is a cosine or sine function with ω_{shift}.

By choosing a proper ω_{shift}, one component of V_{Input} will be resonant with the cavity. In practice, it is better to chose ω_{shift} such that $\omega_{sig} + \omega_{shift} = \omega_0$, since the higher ω_0 suffers less influence of harmonics. In this case, $\omega_{sig} - \omega_{shift}$ is always non-resonant with the cavity for non-zero ω_{shift}.

To measure the frequency of the signal, the resonant frequency should be set to a fixed frequency. The reference frequency should be scanned to observe the maximum of the response of the digital cavity. The frequency of the signal is then given by $f_0 - f_{shift}$.

3.2 Phase & amplitude analysis based on digital cavity

From the theory of Fourier transformation, signals can be seen as the combination of a group of sine components. Digital cavity is used to choose one of these components and amplify it. With an N_c high enough, other components can be eliminated. Therefore, the output of the digital cavity can be seen as a standard sine waveform containing n points from which the amplitude and the phase can be calculated.

Suppose P_1 and P_2 are two points of the digital cavity output:

$$\begin{cases} P_1 = Asin(\phi + n_1\theta) \\ P_2 = Asin(\phi + n_1\theta + n_2\theta), \end{cases} \tag{3.5}$$

where A is the amplitude of the the output, ϕ is the initial phase, θ is the phase difference between adjacent points which equals $2\pi/n$, $n_1\theta$ indicates the phase difference between the P_1 and initial point, $n_2\theta$ indicates the phase difference between $P2$ and P_1.

From Equation (3.5) we get

$$\begin{cases} \phi = arctan(\dfrac{P_1 \cdot sin(n_2\theta)}{P_2 - P_1 \cdot cos(n_2\theta)}) - n_1\theta \\ A = \dfrac{P_1}{sin[arctan(\dfrac{P_1 \cdot sin(n_2\theta)}{P_2 - P_1 \cdot cos(n_2\theta)})]} \end{cases} \tag{3.6}$$

In practice, several groups of points can be used to calculate the phase and amplitude respectively. The average of the different calculated values gives a better estimation of the phase and amplitude.

3.3 Conclusion

In this chapter, we presented the basic principle of digital cavity. It has several advantages compared to the lock-in amplifiers in the market nowadays. The digital cavity algorithm requires no reference signals when analysing the frequency of the signal, which can be hard to obtain in some cases. The most important point is that the digital cavity algorithm can achieve arbitrary precision in theory. However, in

practice, there are still some limitations to achieve extremely high precision, which will be tested and explained in following chapters.

Chapter 4

High precision signal measurement using digital cavity

In the previous chapter, the generalized LIA algorithm has been introduced and in this chapter, the implementation of generalized LIA is presented. Secondly, an experiment is conducted in order to verify the generalized LIA algorithm. Thirdly, some examples that use generalized LIAs to measure typical physical quantities with extremely high precision such as frequency, phase, amplitude are presented to show the effectiveness of implementation of the technique.

4.1 Software implementation of the generalized lock-in amplifier

The architecture of the experiment system using a generalized lock-in amplifier is shown in Figure 4.1 while the data flow of the generalized amplifier is shown in Figure 4.2. The Generalized Lock-in Amplifier contains two parts, one that scans the

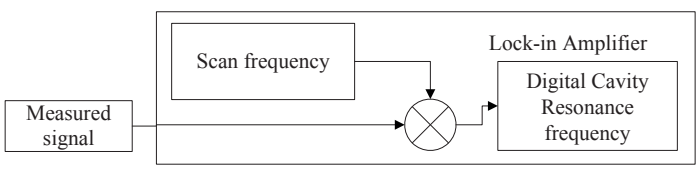


Figure 4.1: Architecture of generalized LIA.

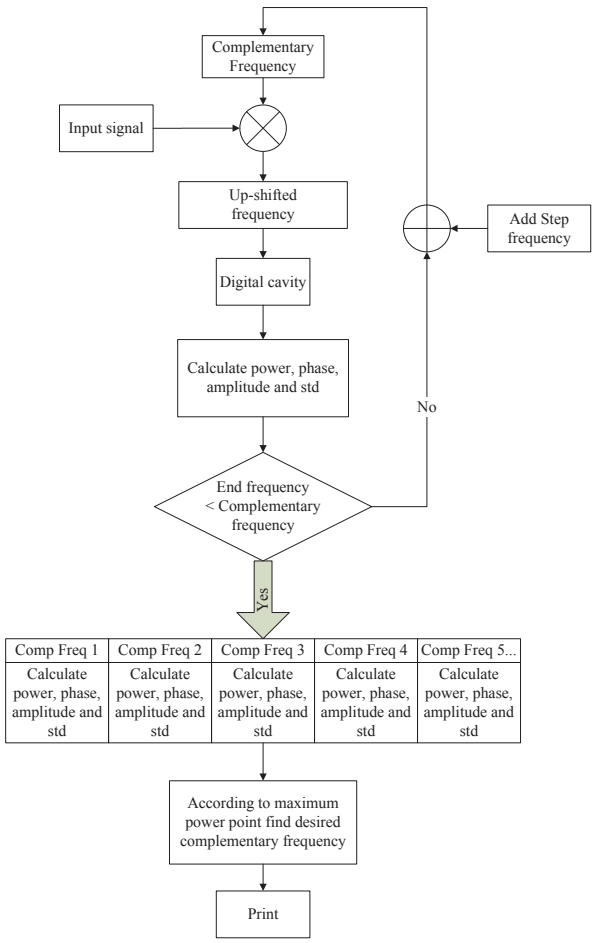


Figure 4.2: Data flow of implementation to generalized LIA.

frequency and one that applies the digital cavity. The frequency scanner is used to generate a complementary frequency. Firstly, the scanner generate a complementary sine signal whose frequency is dependent on the start scan frequency set by the user. Multiplication of the input signal with the complementary sine signal results in that it is being up-shifted to the frequency that is close to the resonance frequency of the cavity. When the up-shifted signal passes through the digital cavity, the phase and the amplitude information can be calculated, which are saved together as a package.

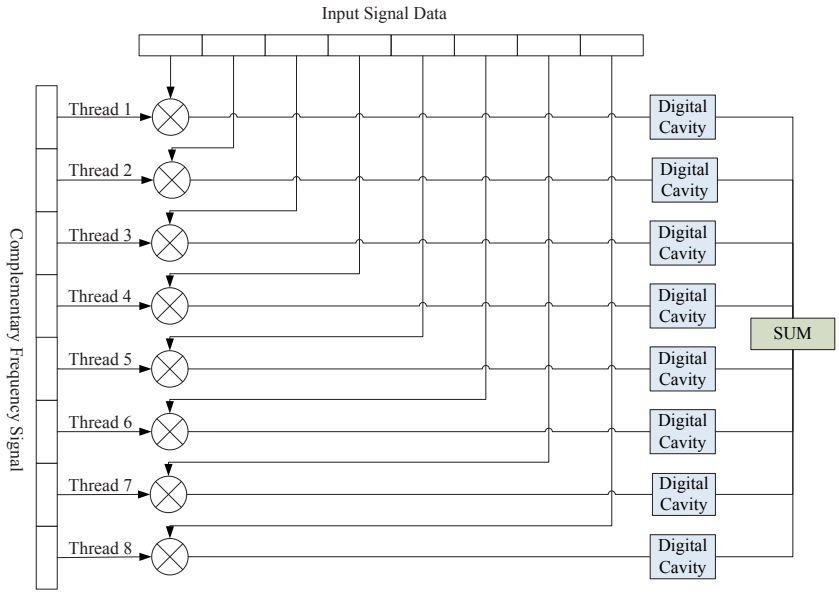


Figure 4.3: Multi-thread of the implementation.

Then the scanner checks the end scan frequency set by the user, which is the upper limit of the frequency scan range. If the current complementary frequency is lower than the end scan frequency, LIA increases the complementary frequency in a small step set by the user and executes the multiplication loop again. If not, it means the scanning is finished. It jumps out of the loop and searches the packages to find the maximum amplitude. This means that at this point the complementary frequency shifts input signal to the digital cavity resonance frequency. The corresponding complementary frequency can be called the desired complementary frequency. As in Equation (4.1), it can use the resonance frequency minus or add desired complementary frequency to get the input signal frequency of the generalized lock-in amplifier.

$$f_{\text{sig}} = f_{\text{resonance}} \pm f_{\text{desired complementary}}. \tag{4.1}$$

To reduce software execution time, multithreading has been used in the implementation of the generalized lock-in amplifier. In total, 8 threads has been used in our design. The number of threads are limited by the maximum number of threads in the CPU. According to Figure 4.3, the input signal data is separated into 8 parts,

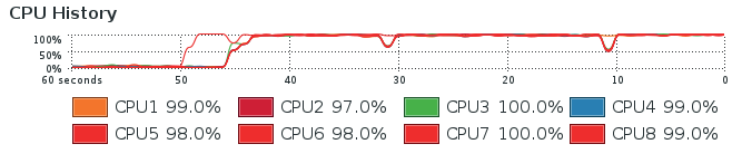


Figure 4.4: CPU utilization.

and the complementary frequency is also separated by 8 so that they match each other. Each part contains $x \times n$ data points, where x is an integer and n is the number of data points per cavity length. The data are simultaneously multiplied and pass through different sub digital cavities concurrently. Finally, the sum of all the responses of the sub digital cavity gives the final response of the complete digital cavity. The speed of the software increases almost 7 times compared to when using a single thread. As shown in Figure 4.4, all the eight threads are fully occuppied. This multithreading technique can be adopted to analyze frequencies beyond the system clock frequency of the DSP processor. With the most advanced commercially available on-board processors and digitizers, signals up to 60-70 GHz can be analyzed using the generalized LIA.

In reality, the way of scanning discussed above can not find the desired complementary frequency in a short time. When the fold number is 10^8, calculations for one individual complementary frequency requires around 30 seconds, which means if 1000 steps of scanning are applied, it will take 8 hours. In practical experiments, scanning 10^6 steps is commonly required, which will take 507 days. Thus, a new scanning method is used in the software implementation. According to the previous method, frequency increases from start frequency to end frequency with a fixed step. If the start frequency is 1 Hz and the end frequency is 1000 Hz, step size of 1 Hz, it needs to sweep 1000 times, that is

$$N = (f_{end} - f_{start})/f_{step}. \tag{4.2}$$

.

New method is a form of step changeable scanning. Figure 4.5 shows the details of the approach to sweep the frequency and the block of 'Original GLIA Sweep by contemporary step frequency' in the Figure 4.5 is the GLIA used the old method(see Figure 4.2), that means the new method is based on old method. The basic idea of this new method is using a larger step frequency to scan the spectrum first, to find a

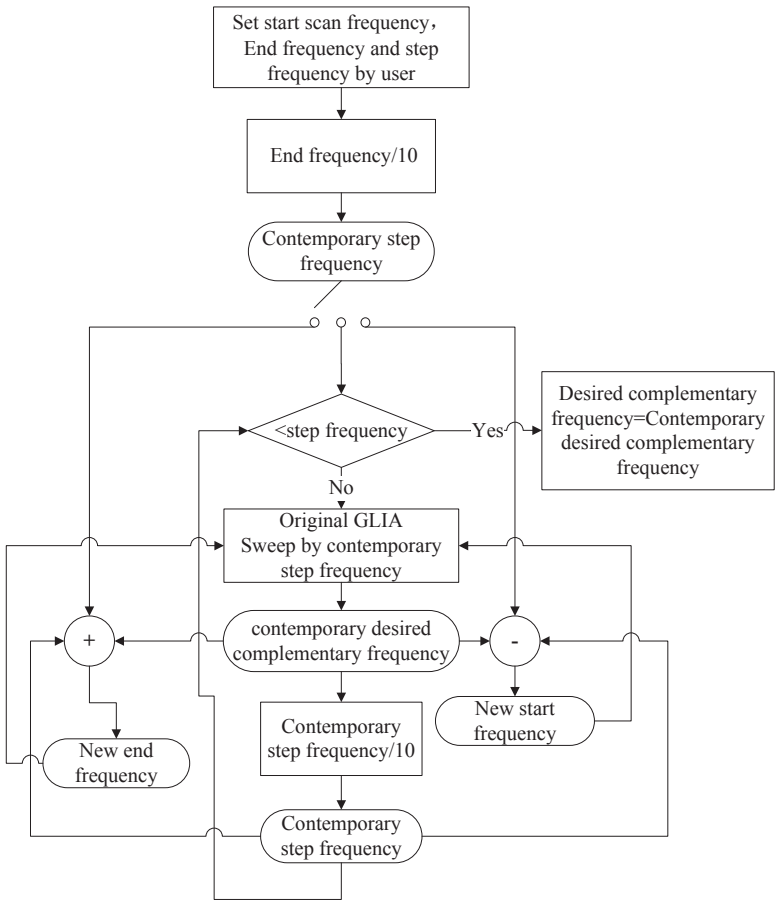


Figure 4.5: The data flow of the improvement scan method.

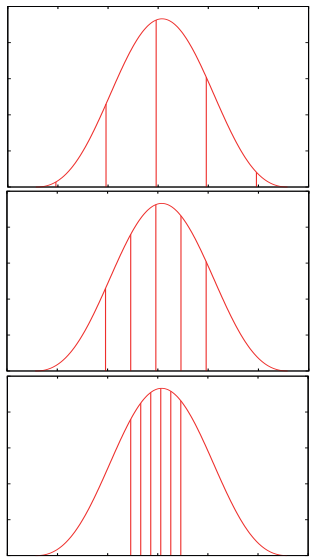


Figure 4.6: New sweeping method.

rough max power point. Then the neighbouring frequency range of the max power point are scanned with a refined step frequency, to make the result more precise. These steps can be repeated several times, according to the precision required and Figure 4.6 shows this procedure.

In the new method, it only needs to sweep about 60 times to find the right frequency, that is

$$N = \log((f_{end} - f_{start})/f_{step}) * 20, \tag{4.3}$$

where N represents the times of scanning. Figure 4.7 shows the relationship between the scan frequency range and the times of scanning. The X-axis represents the numbers of the scanning steps while the Y-axis represents the number of times the digital cavity needs to calculate for finding complementary frequency. The figure 4.7a represents the previous method and the figure 4.7b represents the new method. The figure 4.7c represents the difference scan times between old method and new method. Comparing these two methods, it can be seen that the new method efficiently decreases the times of the calculation. In 10^6 times scanning, it only needs 120 calculations using the digital cavity. With fold number of 10^8, it only needs 1 hour rather than 507 days.

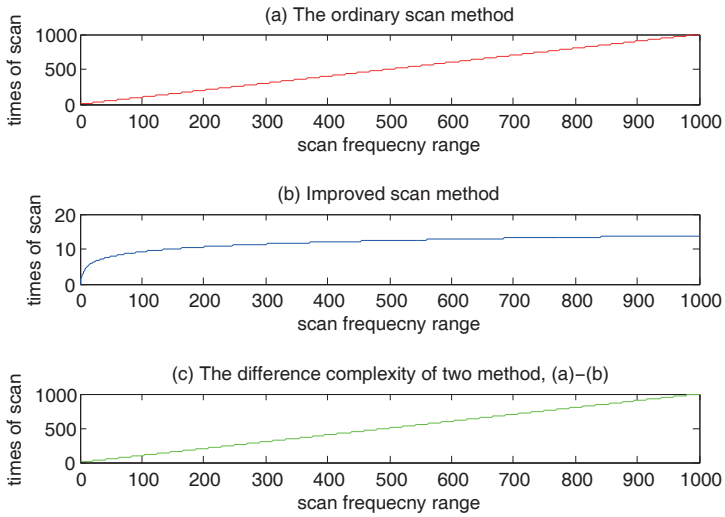


Figure 4.7: The comparison of two methods for scanning.

4.2 Verification of generalized lock-in amplifier

In the previous chapter, the algorithm for the generalized lock-in amplifier has been introduced and an algorithm for software implementation has been developed. In this section, the implementation model is verified by measuring a 245 MHz sine signal generated internally from a computer without environmental noise, and the relationship between precision and Q-factor as well as system wordlength is investigated.

4.2.1 Experiment setup

Figure 4.8 shows the schematics of the experiment. Here we apply generalized LIA to a computer generated signal. If the response from the Lock-in Amplifier is exactly the same as the signal generated by the computer, it will verify that the system can be realized digitally. The advantage of using an ideal signal is that it is directly generated inside the computer and the generalized Lock-in Amplifier is also realized by software so that no environmental interferences is added to the input signal. All interferences come from computer system errors, such as the numerical precision of the numbers represented in system and the precision of some functions.

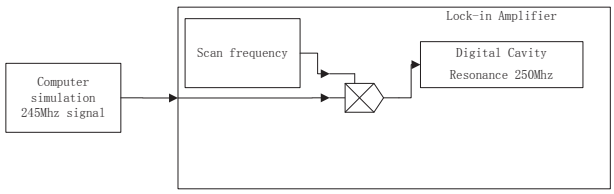


Figure 4.8: Ideal signal test model.

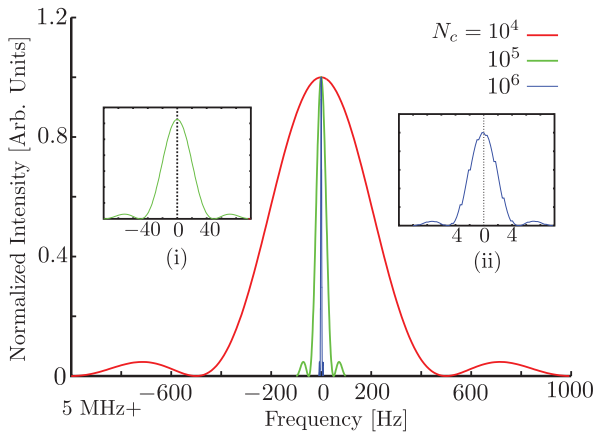


Figure 4.9: Result of the ideal signal test with low fold numbers [5].

In this experiment, as shown in Figure 4.8, computer generates a 245 MHz signal. The scanner generates a 4-6 MHz complementary frequency signal in order to search the desired complementary frequency. The resonance of the digital cavity is set to 250 MHz.

4.2.2 Analysis

The result of the experiments are shown in Figure 4.9 and 4.10. The X-axis in Figure 4.9 and 4.10 represents the frequency response of the complementary signal. N_c represents different fold numbers. The response of the digital cavity using the computer with 32-bit processor in the desired complementary frequency of 5 MHz

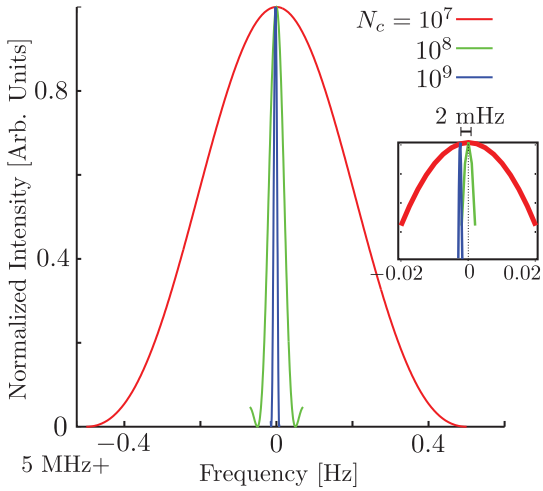


Figure 4.10: Result of the ideal signal test with high fold numbers [5].

for different values of N_c is shown in Figure 4.9. The insets (i) and (ii) show the response with $N_c = 10^6$ and 10^7 in detail, respectively. When N_c is 10^7, the response has a slightly shift towards lower frequency, about 0.2 Hz. This shift is due to the accuracy of the floating point representation as well as the finite accuracy of the sine and cosine functions generated by the computer system. It deserves to notice that this error in precision is inherited in all of the DSP systems due to the finite accuracy of floating point system. Our tests show that using an Intel(R) Core(TM) i7-3770K CPU, gcc compiler (version 4.7) and 32-bit floating number system, we can achieve an accuracy similar of the digital LIA found in the market, which is about few tens of a ppm (parts per million, corresponding to a Q-factor of about 10^5) (Figure 4.9).

As shown in Figure 4.10, few tens of ppb (parts per billion, corresponding to a Q-factor of about 10^8) has been achieved by using double precision. In this case, it can be seen that the artifacts caused by trigonometric function and inaccuracy of the number system at $N_c = 10^9$ when the cavity length n is 20. Here, the error in precision is shown as the shift in the maximum response of the cavity to lower frequencies by about 2 mHz, as shown in the inset (see Figure 4.10). As far as we know, the Q-factor of 10^8 reached with the generalized lock-in amplifier is far better

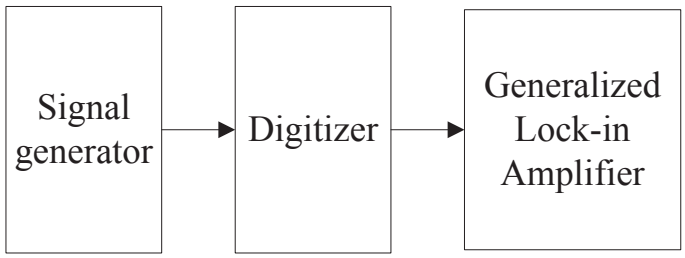


Figure 4.11: The model of high frequency signal measurement.

than commercially available digital LIAs. This makes the generalized LIA not only effective for analysing high frequency signals with high precision but also for any signals that modern ADCs are able to digitize.

4.2.3 Conclusion

With the Q-factor increasing, the frequency artifacts caused by the accuracy of the system sine and cosine function and system word length becomes more obvious. If the Q-factor is higher than 10^6, the artifact is in the 0.1 Hz range using a signal at 5 MHz under 32-bit word length. After increasing to 64-bit, the artifact apparently decreased to around 0.1 mHz (Q-factor 10^8).

4.3 Applications of the generalized lock-in amplifier

In previous sections, the digital cavity algorithm has been tested during ideal situations. The signal being measured was generated in software, which means very low noise is contained. In following sections, the digital cavity algorithm will be applied in real situations to test its performance.

4.3.1 High precision measurement of the frequency

The experiment is shown in Figure 4.11 which is similar to Figure 4.8, but this time the signal is generated from a real signal generator. In this experiment, the

Figure 4.12: Signal generator HP-8656B from Hewlett-Packard.

main purpose is to use a generalized LIA to analyse the frequency of a real signal at high frequency with extreme precision.

Experiment setup

In this test, a 245 MHz signal is generated using HP-8656B signal generator from Hewlett-Packard(shown in Figure 4.12). An 8-bit digitizer ATS9840 from Alazartech (shown in Figure 4.13) is used to digitize the signal at the rate of 1 GS/s (giga sample per second). The digitized signal is transferred to the generalized lock-in amplifier in the computer through a PCI-e interface. The digital cavity of the lock-in amplifier is set with cavity length $n = 4$, for which the resonance frequency is $f_0 = 250$ MHz. The complementary frequency is set to around 5 MHz.

Analysis

Figure 4.14(a) shows the signal acquired by the digitizer. The signal looks more like a triangular wave rather than a sine, which is caused by the insufficient sample rate. This phenomenon is very common in digital sampling systems, and its influence on the response of digital cavity will be tested. Figure 4.14(b) shows the response of the digital cavity around 245 MHz. According to Figure 4.14(b) and (c), the digital cavity shows that the frequency of the signal being measured is 245.0029 MHz, which has a 2.9 kHz difference between which is set in the signal generator. This difference is because the clock speeds of the signal generator and digitizer do not match. In this case, the clock inside the digitizer is slower than that in the signal generator, which causes the frequency difference. The frequency

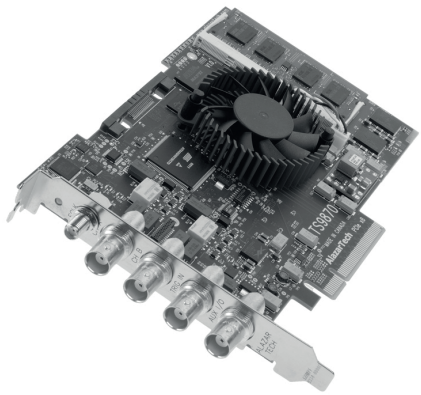


Figure 4.13: Alarzar digitizer ATS9840.

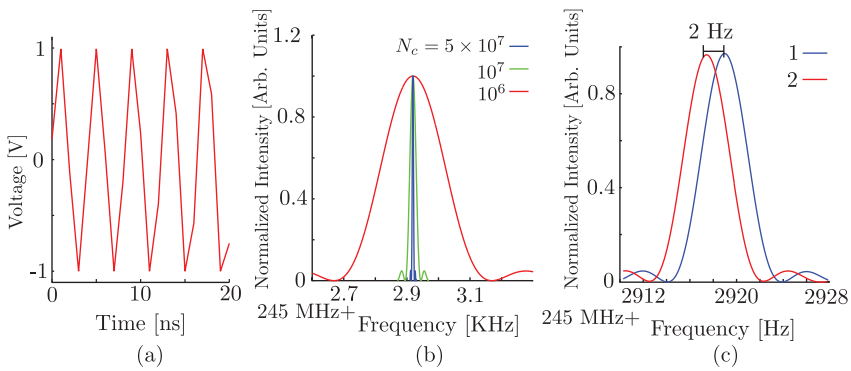


Figure 4.14: The response of high-frequency signals measurement using generalized lock-in [5].

difference between the analysis result and the reference is linearly proportional to the clock frequency between the digitizer and the reference. Actually, this feature of digital cavity enables a promising application, clock synchronisation. When such a frequency difference between digital cavity response and the reference is found, one clock should be tuned until this difference disappears, then the two equipments will work synchronously.

The red and green curves shown in Figure 4.14(b) is the responses of digital cavities with a fold number of 10^6 and 10^7. This result is close to an ideal response which is obtained in previous sections. But when N_c(Q factor) increases to 5×10^7, a wobbling of Hz level can be observed, as shown in Figure 4.14(c). Since in this experiment, 64-bit word-length has been adopted, so hertz level wobbling is not an artifact similar to that of previous experiments, because we have proved that the artifacts should be at milli-hertz level. This 2 Hz shift is the influence of different sample sets. When the fold number of digital cavity is set to 5×10^7, it will require 5 times more samples than that when fold number is 10^7. The instability of the signal generator during the period generating these samples, such as temperature drifts, vibrations are probable factors cause this wobbling.

Conclusion

This experiment shows that the generalized LIA can measure a high frequency signal with high accuracy successfully but we note that wobbling always exists due to the frequency fluctuations caused by environmental factors. We also point out that the difference in the clock speed between the signal generator and the digitizer shows up in the measurements as the shift in the frequency between the generated frequency (in the signal generator) and the observed frequency (in the digital cavity).

4.3.2 Extreme precision phase and frequency analysis of gigahertz signals

This experiment is set to demonstrate the possibility of using digital cavities to measure signals with frequencies at gigahertz level. Besides, this experiment also explores the possibilities of measuring other physical quantities such as length and temperature changes based on precise signal analysing.

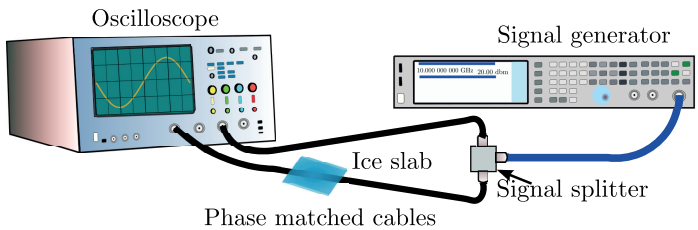


Figure 4.15: Schematics of the setup used to record microwave signals.

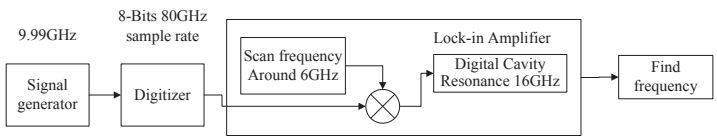


Figure 4.16: The set-up of high precision phase analyze experiment.

Experiment setup

Figure 4.15 shows the set up of this experiment which main equipment includes an oscilloscope, a signal generator, a pair of phase matched cable and a signal splitter. The signal to be measured is generated by an N5183A MXG microwave analog signal generator (shown in Figure 4.17). After coming out of the signal generator, the signal is split by a calibrated signal splitter into two identical replicas. Those are sent to channel 1 and channel 3 of Agilent DSAX93204A Infiniium High-Performance Oscilloscope(Figure 4.18). A pair of phase matched cables, 90 cm long at room temperature. The sampling rate of the digitizer in the oscilloscope is 80GS/s. The digitized signals are temporarily saved in the memory of the oscilloscope. After finishing all of the sampling, the data acquired is transported into a mobile hard disk and then analysed in digital cavity. All of the data is analysed in a PC with a CPU of 8 cores inside. The whole experiment abstract procedure can be found in the Figure 4.16.

$$f_{resonant} = \frac{\text{Sample Rate}}{n}. \tag{4.4}$$

In this experiment, digital cavities with fold numbers of 10^6, 10^7 and 10^8 have been applied. The resonant frequency of digital cavities are set according to Equation (4.4). In this case, the resonant frequency is set to 16 GHz($n = 5$). The

Figure 4.17: N5183A MXG microwave analog signal generator.

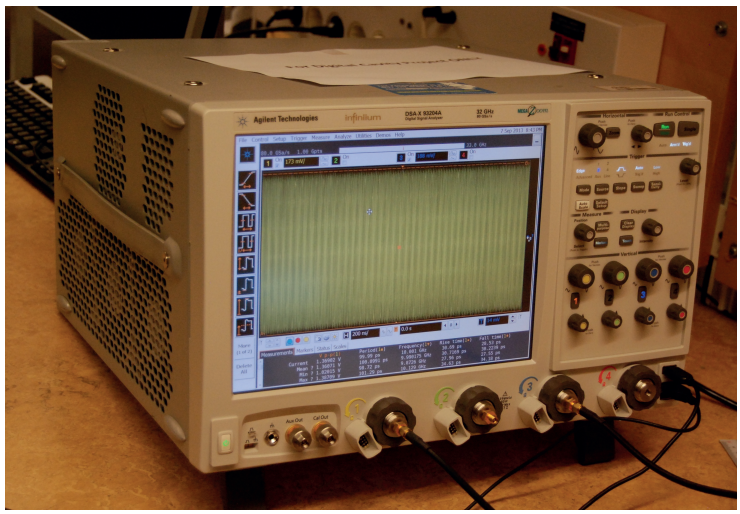


Figure 4.18: DSAX93204A Infiniium High-Performance Oscilloscope.

frequency of the signal being measured is set to 9.99 GHz. So, the complementary frequency scanning range is set to around 6 GHz.

Analysis

Figure 4.19 shows the response of the digital cavity around 10 GHz. With fold number increasing, The full width at half maximum (FWHM) of the response

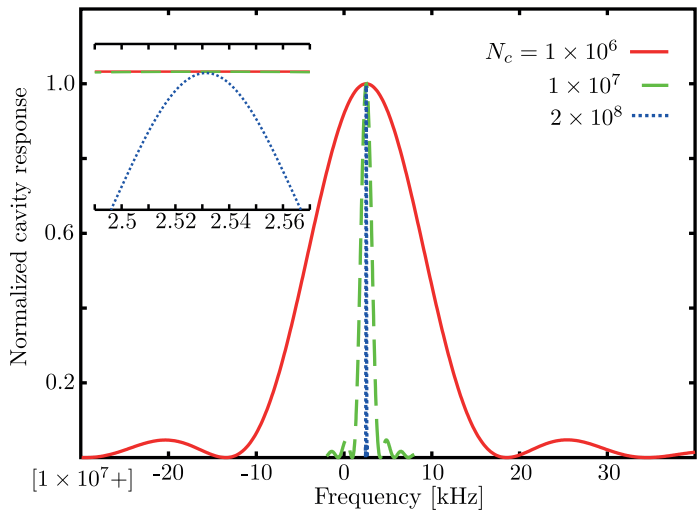


Figure 4.19: Response of the digital cavity to the signal around 10 GHz for different values of N_c.

becomes progressively narrower. The FWHM for $N_c = 2 \times 10^8$(blue line) is about 70 Hz. The maximum power is at $1 \times 10^7 + 2.53$ kHz, which means 2.53 kHz more than the frequency value set in the signal generator. The difference in frequency between signal generator setting value and oscilloscope is caused by the difference in the clock speeds. In our experiments the clock in the oscilloscope lags by 0.25 μs per second of the time lapsed in the signal generator.

The timing precision in the measurements can also be calculated by measuring the average relative phase ($\Delta\Phi$), i.e. the phase difference, between the identical signals acquired by the two channels, as shown in Figure 4.19. The relative phase at the room temperature, after correcting for small drifts, for the signal at 10 GHz is 0.3500 ± 007 degrees. The uncertainty in the measurement can be determined from the standard deviation computed from 10 independent measurements carried out within 10 minutes. The relative phase of 0.35 degrees corresponds to a time difference of 0.1 ps, which is less than the timing tolerance of 25 ps specified for the phase matched cable. The uncertainty of 0.007 degrees corresponds to an uncertainty of 2 fs in the measurement of time.

Conclusion

From the result of this experiment, the relative phase we calculated is below the timing tolerance of the phase-matched cable, which means the result of the measured frequency and the phase is reliable. To some extent, the relative phase we achieve is far more accurate than the one given by the manufacturer, besides it also proves that generalized LIA can measure giga-hertz signals with extremely high precision. It is very attractive to measure microwave signals whose resolution reaches few tens of hertz. This technology can be used on noise rejection, clock synchronization, precision in radar/sonar systems, remote sensing, etc. Besides extreme precision measurement of frequencies, the measurements of amplitude and phase of short wavelength electromagnetic (EM) waves allow us to use the technique to measure other physical quantities with high precision.

4.3.3 Newton's law of cooling by the precision measurement of the phase and amplitude change

After verifying that the generalized LIA is able to measure phase with extreme precision, this experiment will show a practical example of how to use the measurement of the phase and the amplitude to verified Newton's law, besides it also shows that the methodology can be used to calculate the change of the length of the cable caused by contraction. Then it heuristically shows the potential applications of Generalized LIA.

Experiment setup

This experiment shares the same set up with previous experiment. But in this experiment, we will focus on the relative amplitude and phase between the signals in the two cables. The two cables are identical, the only difference is that one of the cables is cooled down by an ice slab during measurements. The data of 25 ms has been sampled at different time nodes, which can be summarized as before cooling, during cooling and after cooling. Before cooling, the temperature of the cable is at room temperature. Then in the cooling process, the temperature of the cable keep reducing from room temperature, until as low as the ice slab. After the whole cooling process, the temperature of the cable stays at the lowest point stably. With the temperature decreasing, the cable will slightly contract, which results in the change of the relative phase between the two cables. This means we

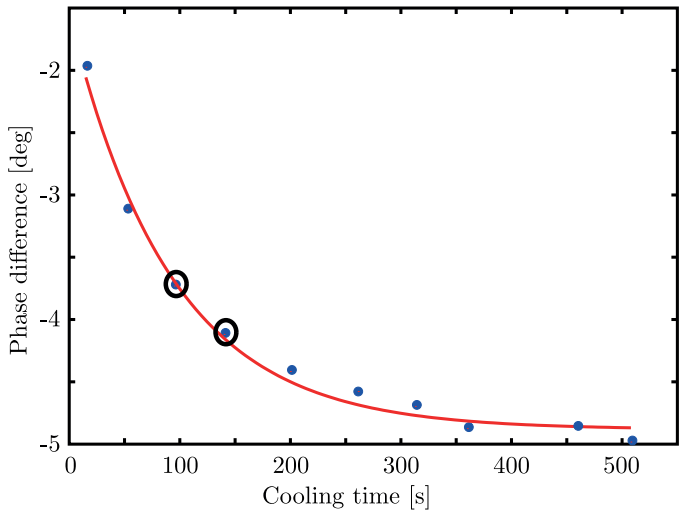


Figure 4.20: Relative phase changes at different time points during cooling.

can measure how much the cable contracted by measuring how much the relative phase has changed. The change in the relative phase, $\Delta\Phi(t)$ as the cooling time t progresses is given by (see Appendix A for the derivation):

$$\Delta\Phi(t) = X_0 + X_1 exp(-kt), \tag{4.5}$$

where k is the cooling constant, X_0 and X_1 are fitting parameters. $\Delta\Phi(t)$ is the difference between the two phases of signals we measured.

Analysis

Figure 4.20 shows the change of relative phase with cooling time. According to Equation (4.5), the relative phase should decrease exponentially as temperature going down. The red curve is the fit of Equation (4.5) to the data points, in which $k = 0.01 \pm 0.01°s^{-1}$ and $X_1 = 3.3 \pm 0.1°$. As the change in the relative phase is proportional to the cable contraction (Appendix A, Equation (6)), k also gives the time constant for the contraction of the cable. The data points used to calculate the length change is marked with circles in the Figure 4.20. For a 10 GHz signal, the wavelength λ_c is 29.979246 mm, which is the wavelength of the EM wave in the

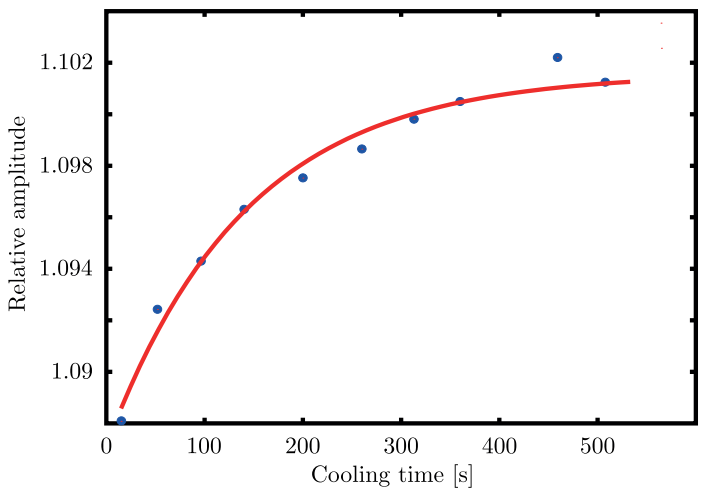


Figure 4.21: The change in the relative amplitude at different times during the cooling of one of the cables.

vacuum. According to the λ_c, it is calculated that the cable has contracted about 32 μm. The λ_c used here is actually lower for the signal travelling in the cable. Hence, the calculated length change is an upper limit value rather than the real value. Nevertheless, the calculation we have presented serves as a proof of principle measurement of the length change. Likely, the upper limit of the length change possible to measure is decided by the precision of the relative phase. In our case, it is 0.5 μm, which corresponds to a relative phase of 0.007 degrees.

$$\Delta\Phi = -\frac{360^\circ}{\lambda_c}\alpha\Delta T L + O, \tag{4.6}$$

where α is the thermal expansion coefficient , ΔT is the temperature change, L is the length of the cable at the room temperature and O is the offset.

Equation (4.6) has shown the relation between the relative phase, temperature change and the length of the cable. With the value of the thermal expansion coefficient $\alpha = 16.5 \times 10^{-6} K^{-1}$ (for copper), Equation (4.6) gives a precision of $0.04^\circ C$ in the measurement of the temperature change. This result also inspires us, if the temperature is controlled precisely, the speed of the EM signal in the cable can be measured by this technique with extreme precision, which can be another promising

application.

The amplitudes of the signals can provide more information about other physical properties of the cables. The change of the amplitude of the signal shows the change of the resistance of the cables during the cooling process. For most of the wires, the decrease in temperature causes the decrement of resistance, which leads to an increase of the amplitude. The change of the relative amplitude during the cooling is given by the following equation:

$$A(t) = \frac{C_0}{1 + r_0[1 - 0.136(1 - exp(-kt))]}. \tag{4.7}$$

The change in the relative amplitude during the cooling is shown in Figure 4.21. The red curve in Figure 4.21 is the fitting of Equation (4.7) to the data points. In the fitting, $k = 0.008 \pm 0.001$, which represents the rate of change in the relative amplitude. It differs from the result of the relative phase measurement. Considering the linearity in the thermal expansion as well as the electrical resistivity, it is reasonable to assume that the rate of change calculated from amplitude measurement differs by about 20% from that in the phase measurement. Besides, the measurements are conducted in an open system, while the result should be much more accurate in an environment under extreme control.

Because the measurements have not been fully calibrated, the physical properties like the speed of the signal in the cable or resistivity of the cable are not brought into the computations. But in principle, the measurements has shown that the phase and amplitude of the signals can be directly used to calculate such quantities. The limitation of this measurements is the digitization noise, namely noise in the signal generator and fluctuations in the environment. If the environment is better controlled, the precision can be improved significantly. If assuming a clean signal is provided and the environmental fluctuations are negligible, the uncertainty in the phase measurement due to the digitization error alone using 8-bit digitizer is at most 10° per cycle of the signal. Averaging over 10^8 cycles reduces the uncertainty to 0.001°. Apparently the uncertainty in the phase 0.007° in our experiments is influenced by sources of signal rather than the digitizer itself. More work can be done to reduce the noise from environmental fluctuations and the source of the signal.

Conclusion

From this experiment, the frequency is measured by generalized LIA with extreme precision. From the frequency a lot of useful quantities can be deduced such as phase, amplitude, resistivity of the cable and speed of signal. The potential capability of generalized LIA can be utilized in different fields. In the future the measurement of higher frequency signals will be possible with development of high-speed digitizers. The technique in this experiment would be highly relevant to systems such as radar, telecommunication, remote sensing, etc. One of the areas where extreme precision measurement of microwaves is likely to have scientific importance is in accelerators that use microwaves to accelerate particles to relativistic velocities.

4.3.4 Signal recovery from noisy background

The possibility of the digital cavity being used as a lock-in amplifier has been presented in previous chapter. In this experiment, we verify the application of digital cavity in signal recovery from a noisy background. To this end, we setup an experiment to recover a 30 MHz signal from a very noise background and measure the response of the photo-diode for signals with various frequencies.

Experiment setup

In this experiment, a 14-bit digitizer ATS9440 from Alazartech, a He-Ne laser (25LHP121-230) from CVI Melles Griot and two acousto-optic modulators (AOM) are used. The digitizer is a different model from the previous experiments. They are both manufactured by Alazartech, share the same features, except the precision of sampling. Here we use a 14-bit digitizer.

The experimental setup is shown in Figure 4.22. In the measurements, the light from the source is attenuated by a neutral density filter, ND in the figure. Then it is split into two beams using a beam splitter (BS1). This splitter is made to produce two beams with identical phase and power. The phase of two beams are modulated by the AOMs. The modulation will cause the intensity modulation of the combined beam at the difference frequency of the phase modulation. The phase modulation frequencies of the two AOMs are carefully set to make the difference frequency of upto 30 MHz. Then the two beams are recombined at the beam splitter BS2. The combined beam is sent to the photo-diode to be transformed into the electrical

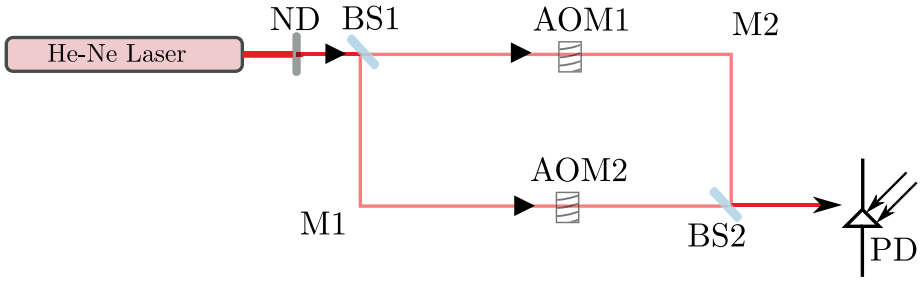


Figure 4.22: Schematic of the setup for recording the noisy data using photo- diode (PD) [5].

signal. This electrical signal is sampled by the digitizer at a sample rate of 50 MS/s (mega samples per second). The data acquired is stored in a file and analysed by digital cavity software.

Result analysis

Figure 4.23(a) and (b) are the original signal acquired by the photo-diode, their frequency is manually set to 50 kHz and 4 MHz, respectively. These two figures show the response of the photo-diode signals with different frequencies. It is obvious that the signal is quite clear at low frequency and when the frequency of signal goes higher, it is severely affected by noise. The reason is that the amplifier used in photo-diode usually has a cut-off frequency. The cut-off frequency can be seen as the upper limit frequency that the photo-diode will work properly. Performance of the photo-diode will fall down rapidly for signals beyond the cut-off frequency. The response of photo-diode will be very noisy at those frequencies. This effect is shown in Figure 4.23(e).

Figure 4.23(c) and (d) show the response of the digital cavity to the signals around 50 KHz and 4 MHz, respectively. The resonant frequency of the digital cavity used here is set and fixed to 5 MHz, n is set to 10 and N_c is set to 2000. Frequency scanning is done by multiplying the recorded data with a sequence of sine waves of different ω_{shift}. By changing ω_{shift}, only one frequency component will be chosen, which make the product of the multiplication match with the resonant frequency. For example, for 50 KHz signal, ω_{shift} should be 4.95 MHz in this case, and for 4 MHz signal, the ω_{shift} should be 1 MHz instead. The signal at 50 kHz has little noise. Consequently the response of the digital cavity is also close to ideal noise

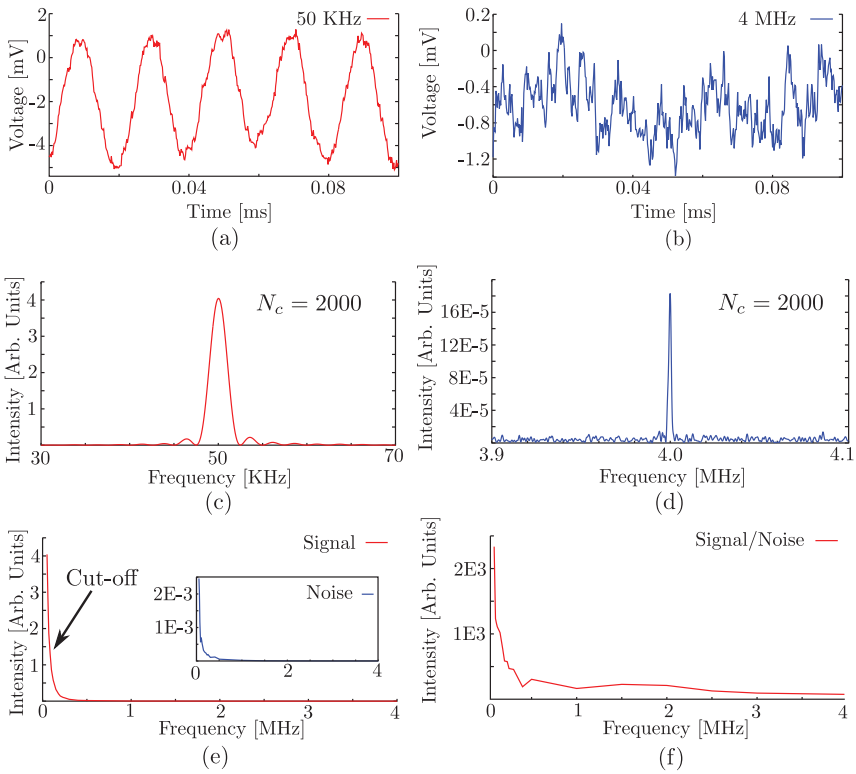


Figure 4.23: Signals with different noise backgrounds and the corresponding responses of the digital cavity [5].

free signal, as the result from previous experiments. However, for 4 MHz signal, the noise is also presented in the response of digital cavity around the central frequency. The response of the digital cavity around the resonant frequency is taken as the measure of the noise. For example, the average of the response of the digital cavity for the frequencies 3.9-3.95 MHz and 4.05-4.1 MHz in Figure 4.23(d) is taken as the noise in the signal for 4 MHz after applying the cavity.

Figure 4.23(e) shows the amplitudes of the response of the digital cavity for photo-diode signals at different modulation frequencies. The amplitudes of the signals at higher frequencies are negligible compared to the signals at the lower fre-

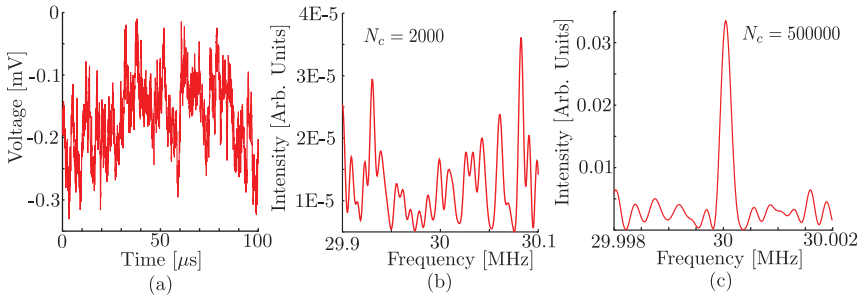


Figure 4.24: Response of digital cavity with different N_c [5].

quencies. Nevertheless, the signals at high frequencies are easily discernible from the noise. As shown in Figure 4.23(f), the noise at the lower frequencies is about 3 orders of magnitude smaller than the signal after the application of digital cavity, while it is about 80 times smaller in the case of the higher frequencies.

As mentioned previously, digital cavity can be used as lock-in amplifier to extract the signal from the noise. Figure 4.24 show the result of this application. Figur 4.24(a) shows a small section of the original signal, it is sampled at a sampling rate of 100 MS/s. The desired signal is at 30 MHz, but is covered in background noise and hard to recognize from the figure. Figure 4.24(b) and (c) show the response of two digital cavities with different N_c. In Figure 4.24(b), resonance frequency is set to 25 MHz, and N_c is 2000. In this case, signal is down-shifted 5 MHz to match the resonance frequency. Then another digital cavity with higher N_c, 500000, is applied to the same signal and its response is shown in Figure 4.24(c). In this figure, the signal is clearly distinguished from the background noise. This result shows that digital cavity works well as a lock-in amplifier.

Chapter 5

FPGA-based implementation for real-time application

In previous chapters, the software implementation has been introduced. However, the software implementation is not a suitable solution for real-time measurement of high-frequency signals. For example, the digital cavity still needs to spend 1 hour to analyze a signal with a fold number of 10^8. The processing time should be decreased in some time-urgent applications. If digital cavity is implemented in hardware with a suitable architecture, the real time request can be fulfilled. Actually, when it is implemented in hardware, the key challenge is the implementation of a high-speed and accurate sine wave generator at reasonable cost. In the software model, the input phase and output value of sine generator is 64 bits double float numbers. From the sight of hardware, the longer wordlength costs more on-chip resource and decreases the global clock frequency. Thus, it is unavoidable to find a trade-off way to decrease the bits-width. This chapter first discusses three methods of sine wave generation. After the error and complexity of these three methods are analyzed, a suitable method for our implementation is given. Moreover, the architecture of implementation on XUPV5-LX110T FPGA platform is introduced. After that, a practical experiment is performed by analyzing the frequency of a real-life signal using the implemented digital cavity and make comparison between the hardware result and software simulation.

5.1 Theoretical foundation for sine wave generation

This section mainly focuses on the comparison of possible solutions, which are CORDIC algorithm, Taylor series interpolation and look-up table for the sine generation, because this is the critical challenge of the digital cavity implementation. It is required to generate high precision sine waves reasonable hardware cost.

5.1.1 Sine wave generation algorithms

CORDIC

The CORDIC algorithm was first presented by Jack E. Volder in 1959 [6]. It is an algorithm using iterations to calculate hyperbolic and trigonometric functions. The most important advantage of this algorithm is a hardware multiplier is not required. Only addition, subtraction, bit shift and look-up table are needed. The CORDIC algorithm is initially used to rotate a vector to another vector according to a certain angle. e.g., rotating vector (x_i, y_i) with a_i to form a new vector as (x_{i+1}, y_{i+1})

$$x_{i+1} = x_i \cos\alpha_i - y_i \sin\alpha_i, \tag{5.1}$$

$$y_{i+1} = y_i \cos\alpha_i + x_i \sin\alpha_i. \tag{5.2}$$

Equation (5.1) and (5.2) can be further transformed [7] to

$$\begin{aligned} x_{i+1} &= x_i \cos\alpha_i - y_i \sin\alpha_i \\ &= \cos\alpha_i (x_i - y_i \tan\alpha_i) \\ &= \frac{1}{R_1}(x_i - y_i \times d_i \times 2^{-i}), \end{aligned} \tag{5.3}$$

$$\begin{aligned} y_{i+1} &= y_i \cos\alpha_i + x_i \sin\alpha_i \\ &= \cos\alpha_i (y_i + x_i \tan\alpha_i) \\ &= \frac{1}{R_1}(y_i + x_i \times d_i \times 2^{-i}). \end{aligned} \tag{5.4}$$

If the values of $\tan\alpha_i$ are constrained to be power of two in Equation (5.3) and (5.4), then $\tan\alpha_i$ is simplified to a bit shift and $\cos\alpha_i$ turns to be, $\frac{1}{R_i}$, a scale factor which is a constant operand. d_i is the sign of α_i. For clockwise rotation, d_i is

a positive value and vice versa. This transformation gives that vector rotation only need three adders and two shifters for one iteration without any multiplication. The scale factor can be calculated in advance using Equation as(5.5) [8].

$$R_{i+1} = R_i\sqrt{1+\tan^2\alpha_i} = R_i\sqrt{1+2^{-2i}}. \tag{5.5}$$

Equation (5.3) and (5.4) can also be presented in a matrix format

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = R_\alpha \begin{bmatrix} x_i \\ y_i \end{bmatrix}, \tag{5.6}$$

where $R_\alpha = \frac{1}{R_1}\begin{bmatrix} 1 & -d_i \times 2^{-i} \\ d_i \times 2^{-i} & 1 \end{bmatrix}$. Any angle of sine that is to be calculated can start with $v = (1, 0)$ then rotated by α (counterclockwise)

$$\alpha = \alpha_1 \times d_1 + \alpha_2 \times d_2 + \alpha_3 \times d_3 + \cdots + \alpha_n \times d_n, \tag{5.7}$$

where $\alpha_n = \arctan(2^{-(n-1)})$. The result is

$$V_r = V \times R_\alpha = (((v \times R_{\alpha_0}) \times R_{\alpha_1}) \cdots \times R_{\alpha_n}), \tag{5.8}$$

If the iterations are enough, any sine value can be calculated by Equation (5.8) [8].

The hardware architecture that performs three iteration is shown in Figure 5.1. The sign bit component is decided by α, and it is used to control the add module switch between add and minus operation. The shift length for every iteration is fixed, so the shift component is easy to realize. The precesion of the sine result is proportional to the number of iteration N. Every time the precision of angle increase one bit, one more iteration unit is needed, that is, three adders and two shifters added in hardware and the latency will increase one clock cycle. So the complexity of the hardware is related to the number of iterations or the width of the phase.

Direct look-up table

This is the most direct way to achieve a sine value. All sine responses are pre-calculated and saved in the memory in advance. The values used for approximating a sine wave are stored in a table in the memory as shown in Table 5.1. It is similar to a Hash list, and every phase relates to a sine response. N in Table 5.1 represents the

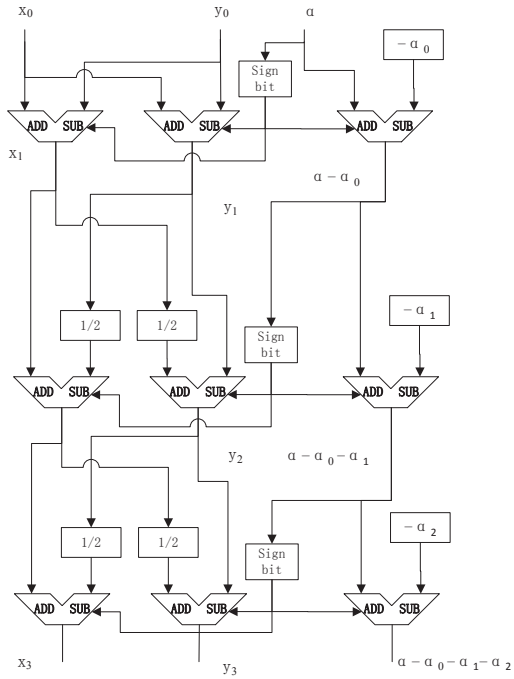


Figure 5.1: CORDIC algorithm architecture on hardware for multiple iterations.

table length and i is the index into the table. All samples are evenly distributed on one sine period. The interval of every two samples is $\Delta(\Delta = \frac{2\pi}{N})$. Increasing table depth N, the precision of sine function can be more accurate, but more memory space is required. Note that the length of the table can be efficiently reduced in that, except for the sign, only one quarter of the table values are unique. Therefore only quarter of table needs to be stored beforehand so as to save the memory. The frequency of the digital sine wave generated depends upon the time interval and the phase angle increment ($\Delta = \frac{2\pi}{N}$) between successive table accesses. If the table is accessed every T seconds, in another word, phase angle increment is equal to one delta. The FTF (Fundamental table frequency) of the sine wave synthesized will be:

$$\mathrm{FTF} = \frac{1}{\mathrm{NT}}\mathrm{Hz}. \tag{5.9}$$

On the other hand, if M is larger than one, for example every second ($M = 2\Delta$)

i=N-1	sin[(N-1)×360/N]
	⋮
i⇒	sin[i×360/N]
	⋮
	sin[5×360/N]
	sin[4×360/N]
	sin[3×360/N]
	sin[2×360/N]
	sin[1×360/N]
Base ADDRESS(i=0)	sin[0×360/N]

Table 5.1: Sine look-up table [8].

or third ($M = 3\Delta$) entry is read, and the table is still accessed every T seconds, then the frequency of the sine wave generated is

$$f = M \times \text{FTF Hz}, M \leq \frac{N}{2}. \tag{5.10}$$

If M is an integer value, only multiples of the FTF can be generated whereas, if delta is allowed to be fractional, any frequency up to the $N/2\times$FTF can be generated. The maximum value of M is $N/2$ because at least two samples per cycle are required to generate a sine wave without aliasing. When fractional values of M are used, points between table entries must be estimated using the table values. The common way to estimate is to truncate the input phase. As shown in Figure 5.2, the phase of samples in look-up table appears at the points of every $2\pi/N$ interval, that is $2\pi/N \times i$, where i is integer. If i is not limited in the set of integer, that means the phase required to locate between two phases entries such as $2\pi/N \times 1.5$. However no such phase can be found in the look-up table. So, a truncation method will be used to estimate the sine result of required phase. $2\pi/N \times 1.5$ is truncated to $2\pi/N \times 1$ which is one entry of look-up table.

The hardware cost is related to the depth of look-up table and the word-length of each element in the table. If the phase resolution increases one bit and noise is required to almost keep the same level as before, the look-up table address needs to increase one bit, that is, the size of RAM will be doubled. Thus, the compensation for increasing the precision of the sine wave is relatively large.

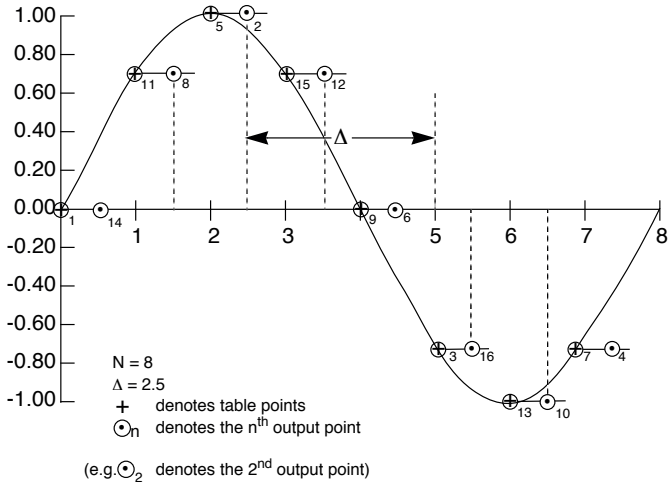


Figure 5.2: Generating sine wave using LUT [9].

First order Taylor series with look-Up table

Taylor series approximation is another method to acquire a high precision sine. Taylor series equation can be given as: [10]

$$f(x) = f(x_0) + f^1(x_0)(x - x_0) + \cdots + \frac{f^n(x_0)(x - x_0)^n}{n!} + o(x - x_0)^{n+1}. \quad (5.11)$$

The function uses a known point x_0 to approximate an unknown point value $f(x)$. In order to increase the accuracy of $f(x)$ calculated by Equation (5.11), two methods can be used. One is to increase the order of the equation that means more entries must be added. The other way is to make x_0 close to x so that $o(x - x_0)^{(n+1)}$ in Equation (5.11) will be close to infinitesimal. The number of multiplications should be reduced as much as possible. Considered the complexity of algorithm and the precision of sine function, one order Taylor series is only discussed here. One order Taylor series can be derived by Equation (5.11):

$$f(x) = f(x_0) + f^1(x_0)(x - x_0) + o(x - x_0)^2, \quad (5.12)$$

Replace $f(x)$ with $\sin(x)$:

$$\sin(x) \approx \sin(x_0) + \cos(x_0)(x - x_0). \quad (5.13)$$

In the equation, $\sin(x_0)$ and $\cos(x_0)$ are calculated in advance and saved in the look-up table. Because sine can be easily transferred to cosine according to periodic feature that is, $\sin(x + \pi/2)$ equal to $\cos(x)$, they can use the same look-up table but only need a $\pi/2$ phase offset. Figure 5.3 shows the method to interpolate value between two entries. The slope of the line segment between successive table entries is $\cos(x_0)$.

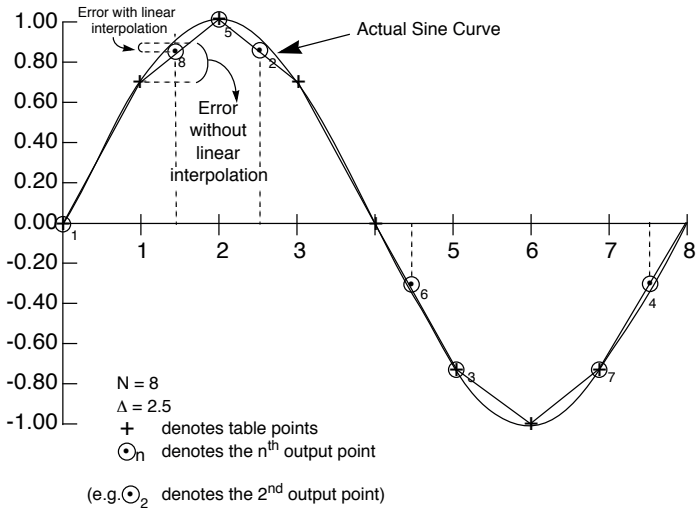


Figure 5.3: Estimation method of first order Taylor series with look-Up table [9].

5.1.2 Precision requirement and complexity analysis in ideal situation

Digital cavity requires high accuracy in sine wave phase resolution. The phase resolution of the sine wave is related to the step frequency of generalized LIA. Step frequency is the frequency resolution of generalized LIA and step frequency is reflected by the phase difference between adjacent points. Thus, frequency resolution can be expressed by minimum phase difference, that is, phase resolution:

$$\alpha \leqslant \frac{2\pi \times f_{\text{step}}}{f_{\text{Sample Rate}}}, \tag{5.14}$$

where f_{step} is the step frequency of the generalized LIA, $f_{\text{Sample Rate}}$ is the sample rate of generalized LIA, and α_{req} is the phase resolution of the sine generator.

For CORDIC rotation, the phase resolution depends on the iteration number since the phase generated by each iteration unit is constantly approaching the target phase as Equation (5.7). After n times of iterations, phase resolution turns to be the minimum term of the Equation (5.7):

$$\alpha_{\text{req}} = \arctan(2^{-(n-1)}). \tag{5.15}$$

Using Equation (5.14) and (5.15), the relationship between iteration n and step frequency and sample rate can be achieved:

$$n \geqslant 1 - \log_2(\tan \frac{2\pi \times f_{\text{step}}}{f_{\text{Sample Rate}}}). \tag{5.16}$$

Assuming that the digital cavity have 1 GHz sample rate and 0.1 Hz step frequency. According to Equation (5.16), iteration number should not be less than 32 times. The maximum value of phase error is equal to phase resolution, that is, 4.66×10^{-10} rad (calculated by Equation (5.15)). The phase error influences the amplitude error of sine. The maximum amplitude error occurs at the phase of 0 and π. Because around 0 or π, amplitude of sine increases rapidly, phase error significantly causes more error in the amplitude of sine.

For the directly look-up table algorithm, phase resolution is decided by the depth of look-up table. The relationship of phase resolution and look-up table depth N can be expressed as:

$$N \geqslant \frac{f_{\text{Sample Rate}}}{f_{\text{step}}}. \tag{5.17}$$

Assuming a measurement of 1 GHz sample rate and 0.1 Hz step frequency, according to Equation (5.17), the depth of look-up table should be no less than 10^{10}. The maximum phase error is equal to the phase resolution. The maximum amplitude sine error appears at 0 or π. Since the amplitude of sine increases rapidly around 0 and π, these points make amplitude error impact greatest by phase error.

For the first order Taylor interpolation, phase resolution is also decided by the depth of the look-up table. But since the first order Taylor interpolation has been used, from Figure 5.3, the amplitude error of the first order Taylor interpolation is far less than direct look-up table. Therefore, under the premise of equal amplitude error, the first order Taylor interpolation can relatively reduce the depth of the look-up table.

The above analysis is under the condition of ignoring the finite word length effect which influences the precision of input/output and intermediate values. Under

ideal situation of 1 GHz sample rate and 0.1 Hz step frequency, CORDIC rotation needs 32 times of iteration. The look-up table needs the depth of 10^{10} and the first order Taylor interpolation needs no more than 10^{10} depth according to the previous analysis. The specific number needs to be found by experimentation.

In the ideal situation, the complexity of the hardware is extremely high. It is inevitable to decrease the complexity of the hardware, at the cost of de-gradating the sine wave precision which will affect the Q-factor of the digital cavity. However, in some applications, the Q-factor of the digital cavity can be relaxed. For example, if the measured signal is a single frequency signal, the error of the response of the digital cavity basically does not affect the ability of the generalized LIA. Next section will analyze how this can be utilized to reduce the complexity of the sine wave generator.

The parameters that affect the complexity of the three methods are summarized in Table 5.2. These parameters directly affect the complexity of hardware. Besides these parameters are also related to the precision of sine wave. Therefore, these parameters must be careful selected before hardware implementation.

Method	CORDIC rotation	Directly look-up table	First order Taylor interpolation
Parameters	Iteration number	The depth and precision of look-up table	The depth and precision of look-up table

Table 5.2: The key parameters relate to hardware complexity.

5.1.3 Experiment to select the key parameters

In hardware design, it is difficult to generate a sine function of high accuracy like the software, because of the limitation of on-board resources, besides the requirement of high-speed operators is also a limiting condition. Therefore, we must consider how to set appropriate parameters for our hardware design. For direct-look-up table, the table depth and samples width must be considered. If the table depth and sample width are set too large, the error and noise would be low but it also cost a great amount of memory resources. For the first order Taylor interpolation with look-up table, the depth and samples width of look-up table must be also considered, although the table depth can be shallower than the direct look-up table. Besides the sample width is related to the multiplier width in the Taylor series. If the multiplier

width is too high, it not only needs more on board multipliers to construct a suitable long width multiplier but also prolong the critical path which will decrease the global system clock frequency. For CORDIC, the latency of output to input is related to the number of iteration. So if the design needs low output latency, the iteration number must be considered. Moreover, the hardware resources are also proportional to the iteration number. Therefore, it is important to choose suitable values for these parameters so as to not only make digital cavity work properly, but also make the design realizable. Apparently, a lot of aspects of reasons for error are generated for each method, and it is complex to give a best analysis conclusion from algorithm of error, therefore we use experiment to find a group of suitable parameters to achieve the best approximation to ideal sine function.

The response of hardware design must be close to the software simulation result by using system sine function as much as possible. Besides, the utilization of on-chip resource must be considered. So we must choose some typical values which are fit to the xilinx Vertex-5 110t to analyze the suitable parameters. The following cases have been choosen for analysis: 10 samples with 32 bits look-up table, 100 samples with 32 bits look-up table, 1000 samples with 32 bits look-up table, 10 samples with 64 bits look-up table, 100 samples with 64 bits look-up table, 1000 samples with 64 bits look-up table, first order Taylor series with 10 samples with 32 bit look-up table, first order Taylor series with 100 samples with 32 bits look-up table, first order Taylor series with 1000 samples with 32 bits look-up table, first order Taylor series with 10 samples 64 bit look-up table, first order Taylor series with 100 samples with 64 bits look-up table, first order Taylor series with 1000 samples with 64 bits look-up table, CORDIC with 8 iterations , CORDIC with 12 iterations, CORDIC with 24 iterations.

The experiment is based on the software model mentioned in Chapter 4. The frequency of measured signal is around 9.99 GHz. It is deserved to mention that the measured signal is one frequency signal. All of the cases will run under the fold number of 10^6. The cavity resonance frequency is set 16 GHz. All other settings are the same.

The result is that the complementary frequency measured by "perfect" sine is the same with the frequency measured by the following case: 1000 samples with 32 bits look-up table, 1000 samples with 64 bits look-up table, first order Taylor series with 100 samples with 32 bits look-up table, first order Taylor series with 1000 samples with 32 bits look-up table, first order Taylor series with 100 samples

	Frequency under the fold number 10^6 (6000+) MHz	frequency under the fold number 10^7 (6000+) MHz	frequency under the fold number 10^8 (6000+) MHz
"perfect" Sine	9.9974855	9.9974773	9.9975355
1000 Samples 32-bit	9.9974855	9.9974773	9.9975355
Taylor,100 Samples 32-bit LUT	9.9974855	9.9974773	9.9975355
CORDIC 12 Iterations	9.9974855	9.9974773	9.9975355

Table 5.3: The same signal measured under the different Q-factor by different methods

with 64 bits look-up table, first order Taylor series with 1000 samples with 64 bits look-up table, CORDIC with 12 iterations, CORDIC with 24 iterations. Although all the above cases can be used to substitute the "perfect" sine, In order to achieve the least hardware utilization only the following cases are considered: 1000 samples with 32 bits look-up table, first order Taylor series with 100 samples with 32 bits look-up table, 12 iterations. From the Table 5.3, all of these cases are under the fold number 10^6, 10^7, 10^8 and the complementary frequency of every case is the same with the complementary frequency measured by "perfect" sine. In another word, all the cases can be used to trade off with "perfect" sine under different fold under accumulation.

Table 5.4 shows the relative characters measured by the corresponding case, the row of "perfect" sine is the reference result with sine function given by system. And the following three rows are the offset value compared with reference result. The amplitude has been normalized. It can be seen that offset of phase and amplitude are almost 10^{-8} and 10^{-6} respectively which can be neglected. Therefore all the cases can be alternatively used to calculate the relative character of the input signal. In short, after the analysis in this section, the probable settings for hardware design have been discovered. For the direct look-up table, it is recommended to set around 1000 samples with 32 bits width. For the first order Taylor interpolation with look-up table, it is better to set as 100 samples with 32-wordlength, that is output

	Frequency (6000+) MHz	Phase (degree)	Amplitude
"perfect" Sine	9.9974855	43.71721156	0.8574032573
1000 Samples 32-bit	9.9974855+0	43.71721156 -0.000000010000001	0.8574032573 +0.0000000878
Taylor,100 Samples 32-bit LUT	9.9974855+0	43.71721156 +0.000000059999998	0.8574032573 +0.0000088166
CORDIC 12 Iterations	9.9974855+0	43.71721156 -0.000000290000003	0.8574032573 +0.0000088187

Table 5.4: Phase and amplitude result calculated by the three sine generation methods.

width can reach to 9 bits. For CORDIC rotation, output width is around 12 bits. These parameters are reference settings, they can be changed according to practical situation [11].

5.1.4 Complexity comparison

	The complexity of the implementation
CORDIC rotation 12 iterations	36 adders and 24 shifters mainly requires the logic units.
Direct look-up table 1000 samples, 32-bit	32k RAM, mainly needs the memory resources
first order Taylor interpolation 100 samples, 32-bit	3.2k RAM with one multiplier the resources balanced requirement

Table 5.5: The hardware complexity for different methods.

According to the parameters in the experiment, the hardware complexity is compared in this section. As shown in the Table 5.5, it requires 36 adders and 24 shifters to organize the main architecture of 12 iterations of CORDIC. More calculation units are needed in this method, compared with other two methods. According to the basic architecture of this method, it is inevitable to achieve better performance by using pipeline or fold technology, but it is compensate for more

calculation units, more complicated architecture and more output latency. The direct look-up table method takes 32k RAM as its main resource usage. Compared to other two methods, it needs most massive block RAM space. This is the potential bottleneck of this method. If the precision requirement increases, a large RAM occupation increment only leads to small increasing of the sine wave frequency purity. First order Taylor interpolation needs one more extra multiplier than direct look-up table in this method, and the calculation units are not so much cost as CORDIC method. Since one multiplier is involved, the output error is revised and turns less, so the block RAM space can be reduced to a great extent compared with direct look-up table method.

Therefore, the CORDIC method is suitable to the hardware platform with plenty of logic units but without any on-chip multiplier. Direct look-up table is suitable to the platform with extremely numbers of block RAM and the first order Taylor interpolation is the most balanced method. It usually fits to ordinary board, especially development board.

5.1.5 Conclusion

Since we want to fully use the on-chip resources so as to achieve a best case, compared these three methods, first order Taylor with look-up table is the best. Because the on-chip resource utilization is more even and it is relatively easy to realize. In this implementation, all the design requires to be implemented on the platform of FPGA development board. So the IP core is the most convenient and effecient way to generate sine. As analysis in section 5.1.3, the most suitable approach for the design based on development board is the Taylor interpolation method, so DDS(Direct Digital Synthesis) IP core has been chosen to generate sine wave, which is based on the first Taylor interpolation with look-up table. Besides, the clock frequency for DDS IP core can reach 550 MHz [11]. The desired frequency of each sine block is at least 125 MHz, so it fully meets the requirement of the design. The output width is chosen as 24 bits and the input phase width is 48 bits, since the maximum width of the phase increment input of DDS is 48 bits. The performance parameters are shown in Table 5.6.

It is not hard to find DDS occupies 3 BRAMs of 18K. As the analysis result, 100 samples look-up table is enough, which only occupied 3.2k of one BRAM. This is because the BRAM utilization is related to the output length. The IP core considers that the output length needs to be fully used. So it expands the look-up table depth.

Character	Value
Output Width	24 Bits
System clock	125 MHz
Noise Shaping	Taylor Series Corrected
Memory Type	Block ROM
Phase Width	48 Bits
Latency	11
BRAM(18k)	count 3
Xtreme DSP	slice count 3

Table 5.6: The performance of sine generator.

If 100 samples look-up table is chosen, the output length to be fully used is only 12 bits [11]. Therefore, this shows that the wordlength we choose is more than enough to simulate the sine function in the software, besides if on-chip resource is limited, it can also be simplified.

5.2 Hardware architecture

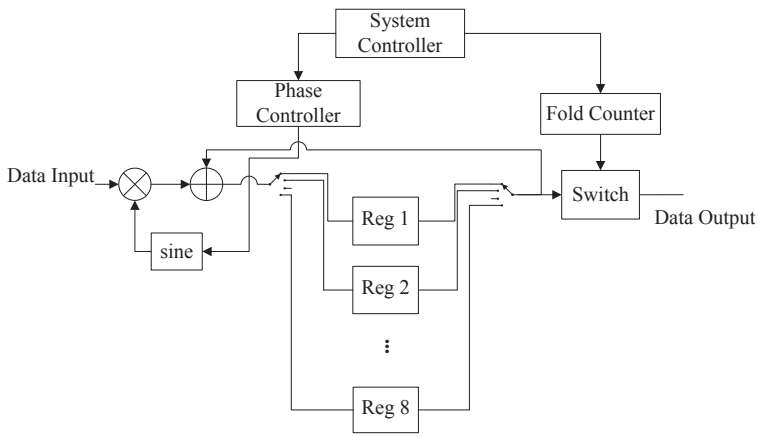


Figure 5.4: The basic idea for architecture of digital cavity.

Without considering the timing constraints of the hardware design, the basic architecture of digital cavity should be like Figure 5.4. Data goes into the digital

cavity at the frequency of sample rate. Every time a data comes, it should be multiplied with the corresponding sine value. The multiplication results will be added up and stored into the corresponding registers. The number of registers equals to the cavity length. After fold number times of multiplication and addition, the results in the registers constitute the response of digital cavity. This is the most direct way to implement the digital cavity. However, in this case, the sine block must generate values at the frequency of sample rate to match the input data. Thereby, it is the bottle neck to achieve very high sample rate. The requirement of our design is to implement digital cavity with a cavity length of 8. Besides, the digital cavity should be able to measure the signal sampled at a rate of 1 GHz, which means that the input rate is 1 GHz. Unfortunately the develop board Xilinx FPGA Virtex-5 XUPV5LX110t only has a global clock running at 500 MHz. It is very difficult to get a output frequency higher than 500 MHz. In this thesis, we developed a parallel structure to deal with the speed problem, whose structure is shown in Figure 5.5.

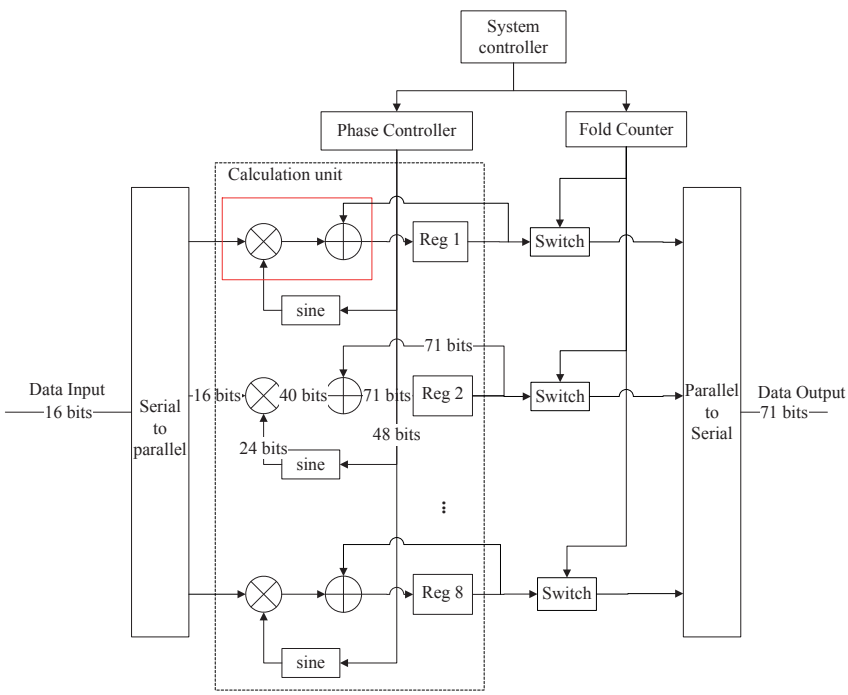


Figure 5.5: Main architecture.

As shown in Figure 5.5, the whole design has several main parts: serial to parallel converter, calculation unit, the phase controller, fold counter and parallel to serial converter. The calculation unit is the core of the design, which consists of 8 individual lanes, according to the cavity length of 8.

Each lane contains an adder, a multiplier, a sine block and a register to store the intermediate result. All the 8 lanes work concurrently. According to the theory of discrete math, the multiplication of two discrete functions can be done by multiplying the corresponding points individually.

In order to match the input data with the sine value of each lane correctly, a serial to parallel conversion block and a phase controller have been added into the design. The serial to parallel converter is used to assign the input data to the 8 lanes evenly, and the phase controller is used to control the phase input of the sine generators.

The serial to parallel converter groups the input data by every 8 samples. They are transferred to multipliers of the eight lanes concurrently. In this way, the order of the original input data is maintained in the same manner as the original input. Thus, the operating frequency of each lane is divided by eight. That means if the sample rate (which is equal to the input speed of data) is 1 GHz, the sine generator of each lane only needs to generate the sine values at 125 MHz. As a result, the whole design is divided into two clock domains. One is before the serial to parallel block, whose main function is to sample the input data. This domain works at the same clock with the ADC. The other clock domain contains the calculation units after the serial to parallel converter. The main function of this domain realizes the algorithm of digital cavity.

An digital synthesiser (DDS) IP core from Xilinx corporation is used here to generate the sine wave. Sine blocks in the calculation unit require two inputs, initial phase and phase increment. By providing an initial phase and a phase increment, the DDS block will generate the value of sine at the initial phase first, increment the phase for each following iteration. Assuming that the 'sample rate' for the input signal is fixed, then the adjacent points on sine wave have a fixed time interval. Therefore, by changing the phase increment, the frequency of output sine wave will be tunable. Another point is the length used to present the phase increment, which will affect the frequency resolution of output sine wave. The increment can be calculated by the following equation: [11]

$$P_{inc} = 2^N \times \frac{f}{SR} \times 2\pi, \tag{5.18}$$

where P_{inc} is the phase increment represented in binary, N is the length to present the phase increment, f is the frequency of output, and SR is the sample rate used to digitize the signal.

The phase controller is designed to provide the phase increment and initial phase for all the sine blocks. The initial phase and phase increment should be configured according to the desired complementary frequency of the digital cavity. The sine blocks generate the sine wave according to these two inputs. As mentioned previously, the speed of the sine wave generation of each sine block should be at 1/8 of the sample rate. Another point is that the initial phases of the 8 lanes should have a phase offset. This phase offset should equal to the 1/8 of the phase increment, which ensures all 8 sine blocks generate a continuous sine wave with complementary frequency.

The fold counter block is designed to save the fold number set by the user and controls the start and stop of calculation. By counting the times addition and multiplication, this fold counter is also used as a result validation unit, telling whether the output is a valid final result or just an intermediate result.

After implementing the design on FPGA, a resource utilization report has been generated and shown in Table 5.7. From Table 5.7, we can see that 50% of DSP48E resource on the FPGA has been used. An DSP48E consists of a multiplier and an accumulator inside. These DSP48Es are occupied by the 8 DDS blocks and the calculation unit. Besides, 432 KB memory has been used. The memory is occupied by the look-up table inside the DDS blocks. According to the report, the system clock can reach 201.694 MHz. This means that this implementation can analyze signals at a sample rate of 201.694 MHz $\times$ 8 $\approx$ 1.6 GHz, meeting the requirement. The critical path of this implementation is from the output of sine generator through the multiplier and adder to the register, which is highlighted by red frame in Figure 5.5. All the bits width of blocks are also signed in the Figure 5.5.

Besides the implementation in Figure 5.6, there are two ways that can enable the digital cavity measuring signals with higher frequency. The first is to pipeline the whole design to break up the critical path by adding a register between the sine generator and the multiplier. The second is to parallel the whole design, as shown in the Figure 5.6. Each digital cavity block is the demo implementation shown in the Figure 5.5. The sample rate equals to the number of paralleled block multiplies the sample rate of each sub digital cavity.

Slice Logic Utilization	Used	Available	Utilization
Number of Slice Registers	3,527	69,120	5%
Number used as Flip Flops	3,527		
Number of Slice LUTs	3,378	69,120	4%
Number used as logic	2,889	69,120	4%
Number used as Memory	80	17,920	1%
Number used as Shift Register	80		
Number of occupied Slices	1,274	17,280	7%
Number of BlockRAM FIFO	16	148	10%
Total Memory used (KB)	432	5,328	8%
Number of DSP48Es	32	64	50%

Table 5.7: Hardware utilization.

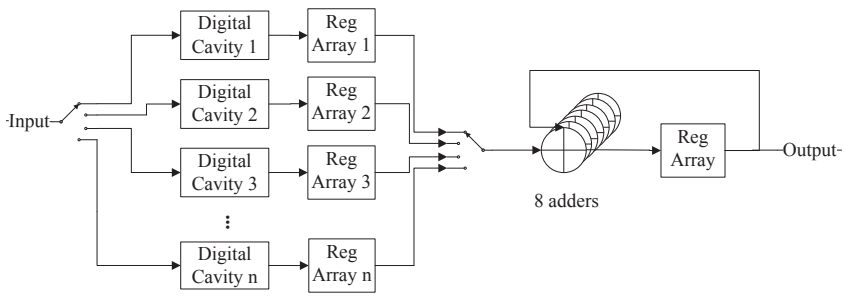


Figure 5.6: Parallel method of digital cavity to increase the sample rate.

5.3 Verification

5.3.1 Environment setup

A hardware accelerator of the digital cavity has been implemented and introduced in the previous sections. In this section, the accelerator will be verified and tested by applying it into a lock-in amplifier, using the Xilinx XUPV5-LX110t FPGA platform.

Two input signals are required for the verification system. One is the sine signal to be measured, and the other is the phase signal used to control the frequency of complementary signal. For the convenience of verifying the design, both input signals are generated inside the FPGA. The architecture of the verification system

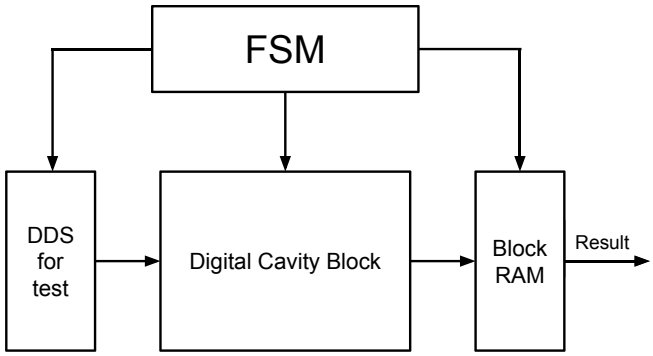


Figure 5.7: Architecture of the verification system.

is shown in Figure 5.7.

Phase control generation

The phase control signal should scan in a customizable range, which is set by the user. In this case, we used 15 points. This means this signal should remain the same during each frequency scan to generate a stable complementary signal and change to another after calculation is finished. According to this requirement, a finite state machine (FSM) is designed, which can be seen in Figure 5.7. A pushbutton on the FPGA is set as the trigger of changing phase input. The whole scanning consists of 15 different phase input, each phase input corresponds with a specific complementary frequency. For each phase input, two states are assigned in the FSM. One state for setting the input, and the other is used as button debounce, which ensure the button works as desired. Figure 5.8 shows the process of state transition.

Input signal

To be tested as a generalized lock-in amplifier, an sinusoidal input signal should be provided as a test signal to digital cavity. In following test, the word length of the sine wave was set to 16 bits, equal to the data format of the Alaztar ATS9780 acquisition board used in the software test. This made it more convenient to compare it with previous results. The bit length to represent the phase increment is set to 48 bits, the upper limit of the DDS block. This length will make the frequency

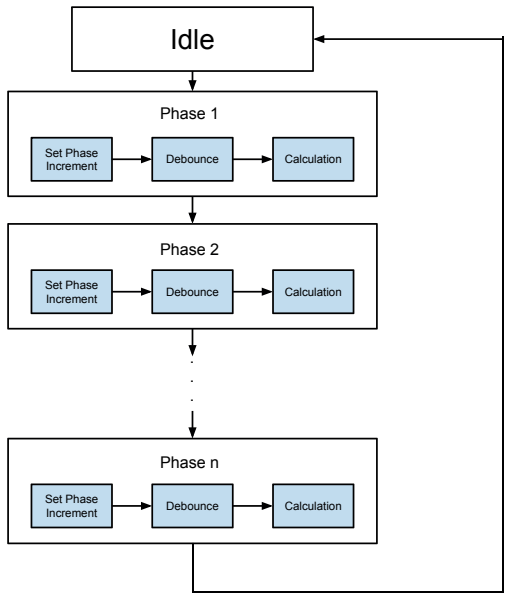


Figure 5.8: State transition.

resolution of the test signal as precise as 0.0001 Hz.

Result collection

To analyze the results of the digital cavity, we used Chip Scope Pro tool. Chip Scope Pro is a powerful tool to analyse the signals inside the FPGA. It is provided by Xilinx as a part of designing tool package ISE 14.6.

During one test, there will be more than ten scanning of frequencies performed. The length of the digital cavity is 8, which means there will be 8 results generated for each scanning. For the whole test, it will be several hundreds, making it very time consuming to collect them serially. To solve this problem, we implemented a RAM inside the FPGA to restore all the results sequentially. When a scan is completed, the result will be sent into the RAM. After all the scans are done, all the results will be extracted from the RAM and transferred to the computers by Chip Scope.

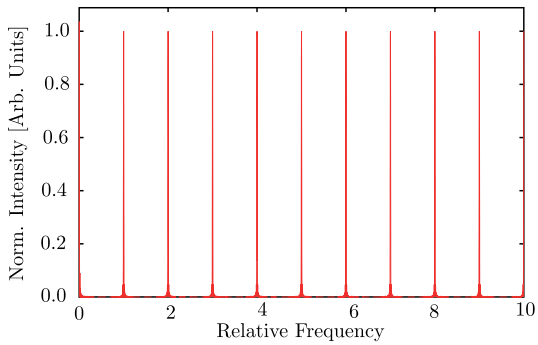


Figure 5.9: Comb filter character of digital cavity [1].

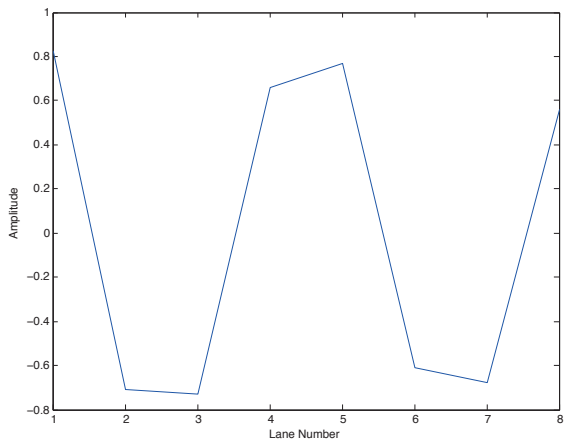


Figure 5.10: Output of digital cavity.

5.3.2 Results and analysis

In the verification, we are testing a signal with a sample rate of 1 GS/s. Then the resonant frequency of this 8 lanes digital cavity is 1000/8 = 125 MHz. A 245 MHz signal generated by DDS block is provided as input. According to the comb filter character of digital cavity, which is shown in Figure 5.9. From the figure, it is easy to see that the pass band of digital cavity is narrowly around integer times of

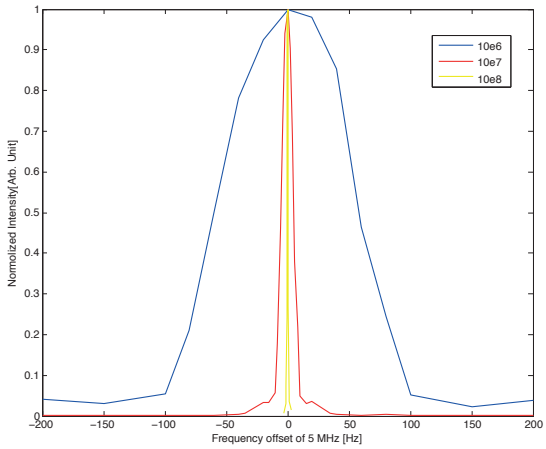


Figure 5.11: Response of digital cavity.

resonant frequency, which means that all the signals with frequencies of integer times of resonant frequency will completely pass the digital cavity. The exact frequency of the signal can be determined by inspecting the output points of the digital cavity.

Since the input frequency is around 245 MHz, the nearest full pass frequency is 250 MHz, which means the complementary frequency should around be 5 MHz. In Figure 5.10, it is easily seen that there are two waveforms. This means that in the fixed time interval, 2 waveforms have passed through the digital cavity, which indicates that the mixed signal is at the 2 times of the resonance frequency, which is 250 MHz. Figure 5.10 shows the output of the digital cavity at 5 MHz when fold number is set to 10^7.

The normalized response of the digital cavity with 10^6, 10^7, 10^8 fold number is shown in Figure 5.11. This figure has got similar results comparing with Figure 4.9. According to the Equation (3.2), the bandwidth of digital cavity with fold number 10^6 is calculated by Equation (5.19),

$$\begin{aligned} 2\xi_{f_0} &\leqslant \frac{2\sqrt{2}f_0}{N_c+1} \\ &\leqslant \frac{2\sqrt{2}\times 125\text{ MHz}}{10^6+1} \\ &\leqslant 353.55\text{ Hz}. \end{aligned} \tag{5.19}$$

and the result is 353.55 Hz. As shown in Figure 5.11, the linewidth of the digital

cavity implemented on FPGA with the same fold number is around 200 Hz. With the fold number increasing 10 times, to 10^7, the bandwidth will simply change to 1/10 of original value, which matches the result shown in Figure 5.11. This proved that the performance of hardware implemented digital cavity meets the prediction of theory.

Chapter 6

Commercialization

Digital cavity is a newly introduced technology, which has great potential and vast development prospects in the commercial market. To commercialize such a new technology, an investigation about the intellectual property, and analysis of potential market have been done in advance. Three schemes of applying this technology into real product are discussed, including a rough analysis of manufacturing costs and human resource requirements.

6.1 Intellectual Property

For the purpose of commercialization of a new technology, intellectual property is the first point being considered. Intellectual property, such as a patent, grant the inventor the exclusive privilege to apply the new technology. Before this master thesis work was initiated, LU Innovation System at Lund University has performed a search in the patent databases and on the Internet in general to determine if the digital cavity technology was new and inventive in relation to prior art in the same field of technology. After a thorough analysis of the documents that was discovered during the prior art search it was decided to pursue a patent protection of the digital cavity technology. An international patent application (also known as a PCT application) with the application number PCT/EP2014/050236 was filed on the digital cavity technology on 8th January 2014.

6.2 Market Analysis

Before launching a new product into the commercial market, a market analysis should be made in order to determine potential competition and potential customers. For digital cavity technology, an important application field is lock-in amplifier. In this field, Stanford Research Systems (SRS) and Zurich Instruments are two markets leading companies. Both companies have a whole range of lock-in amplifiers which can be applied in various situation measurement, but a limiting factor of their products is the frequency range. For SRS, the product with largest frequency range is SR844 RF Lock-In Amplifier, running at the frequency of 25 kHz to 200 MHz. The Zurich Instruments has a product line of ultra-high frequency lock-in amplifiers, with an upper limit of 600 MHz.

However, for products using digital cavity technology, the operation frequency can be much higher than the products mentioned above. This is the most important advantage that the digital cavity has, comparing to other products. This advantage will make the products incorporating this technology very competitive in ultra-high frequency applications, especially beyond 1 GHz.

Digital cavity can also be applied in other fields, such as speed detection using radar. In this field, the most common solution is using DSP as central processing unit. Digital cavity can achieve a much higher precision detecting specific frequencies, which will significantly increase the precision of the radar.

When trying to commercialize the digital cavity technology, there are two options. One is to produce a complete production such as lock-in amplifier, and the other is to fabricate the ASIC only. For a complete lock-in amplifier, universities and researching institutions can be potential costumers. When researching the structure of materials and outer-space radiations, a lock-in amplifier of very high precision can be of great help. For example, in Shanxi University, China, a lock-in amplifier has been used to measure the internal resistance of fuel cells. Their method has been published in Physics Examination and Testing [12].

If the digital cavity is manufactured as a processing core only, companies producing signal measuring equipments can be potential customers. In this way, those competitors mentioned above will instead turn to be potential customers. Besides the SRS and Zurich Instruments, Agilent is also a good customer. Agilent has productions of all kinds of electrical measurement equipments and is a leading company in these technology areas. Digital cavity can perform the specific frequency analysis in much more complex equipment. For these manufacturers of measuring

instruments, digital cavity can significantly improve the precision and performance of their products in signal analysing. On the other hand, manufacturing a complete product will require quite a lot of start up money and negotiations with suppliers of other parts of the product, but fabricating ASICs only will be a much easier solution for a start-up company. This can be a win-win situation for both sides.

6.3 Implementation and Manufacturing Costs Estimation

In order to get a better understanding of the costs associated with implementing and manufacturing, in terms of time and hardware costs, of the digital cavity technology need to be investigated. The first step in the investigation is to determine the different implementation and offering options that exist in order to take the digital cavity technology to the market. Three different ways of implementing and offering the technology have been identified; an FPGA solution, an ASIC solution and an IP-core solution. The final step in the analysis is to determine the best way of taking the technology to the market based on the implementation and the offering of the technology and the market analysis discussed in section 6.2.

6.3.1 FPGA Scheme

Implementing the digital cavity on an FPGA is a solution aimed at manufacturing a complete measurement instrument. This solution is suitable for selling to the end users directly. Since no professional ASIC design tools are required, the design cost of this solution is much lower than others, and the development cycle is shorter at the same time. However, purchasing the FPGAs and ADCs will raise the hardware cost of the product. The work flow of FPGA scheme can be summarized to following steps: hardware logic design, peripheral interface, design simulation & verification, drivers for hardware, UI design of software and documentation. Some software tools are required in the work flow. Matlab is used for modelling, an IDE (Integrated Develop Environment) for logic design, ModelSim for simulation, ISE design suit for the synthesis and place & route of the design, Debussy for debug, Qt for UI design. A rough cost estimation of software tools is shown in Table 6.1.

Another cost that can not be neglected is human resource. Since FPGA solution is developed as a complete product, an ADC interface, a communication protocol

Software Tools	Price
IDE	Free
Qt (UI Design Tool)	Free
ISE Design Suit	Included in FPGA price
Debussy (Debug Tool)	Free
MatLab (Modelling)	5,000 SEK
ModelSim (Simulation Tool)	7,500 SEK

Table 6.1: Design tools cost for FPGA solution.

used to configure from PC platform, a user interface (UI) and proper documentations should be included. For a complete product, the sufficient simulation and verification are quite important to ensure the performance and quality of products. Besides, detailed documentation is also important for end users. Thus, in this scheme, these two steps have been assigned longer period. Human resources required for each steps of the FPGA solution is shown in Table 6.2.

Design Flow	Time required
Hardware Logic Design	2 Weeks
Peripheral Interface	2 Weeks
Design Simulation & Verification	6 Weeks
Drivers for hardware	2 Weeks
UI Design	1 Week
Documentation	3 Weeks

Table 6.2: Human resources required for FPGA solution.

At last, the cost of FPGA solution contains the hardware cost. The hardware cost mainly consists of FPGAs and ADCs. The prices of FPGA and ADC varies with different processing abilities. Thus, the hardware cost for products over 1 GHz and under 1 GHz should be discussed separately. In this case, a Xilinx Virtex 4 ML405 and FMC110 ADC from 4DSP have been chosen in product for under 1 GHz signals. For products above 1 GHz, Xilinx Kintex-7 and 4DSP FMC125 ADC are applied. The prices of FPGAs and ADCs are shown in Table 6.3.

For Lower Than 1 GHz Signals		
Item	Model	Price
FPGA	Xilinx Virtex 4 ML405	11,121 SEK
ADC	4DSP FMC110	10,000 SEK
For Higher Than 1 GHz Signals		
Item	Model	Price
FPGA	Xilinx Kintex-7	19,683 SEK
ADC	4DSP FMC125	21,388 SEK

Table 6.3: Hardware cost for FPGA solution.

6.3.2 ASIC Scheme

If an ASIC scheme is targeted, the digital cavity algorithm will be fabricated into ICs. This will give the products much wider market, and we can focus on improving the performance of digital cavity. ASIC design flow can be divided into front-end design, back-end design, and verification after tape-out. In front-end design, the function of digital cavity should be realized, and verified on FPGA. In back-end design, synthesis, place and route should be done. After that, a post-simulation should be done, to ensure the timing of the circuit correct. Then the design is ready to be fabricated. The last step is to verify that the fabricated sample IC is working properly. ASIC scheme requires a large start-up fund, since the design tools are quite expensive but also necessary. The design costs of ASIC scheme is shown in Table 6.4.

Software Tools	Price
IDE	Free
ISE Design Suit	Included in FPGA price
Debussy (Debug Tool)	Free
MatLab (Modelling)	5,000 SEK
ModelSim (Simulation Tool)	7,500 SEK
Design Compiler (Synthesis Tool, 1 year)	650,000 SEK
SoC-Encounter (Place and Route Tool)	5,000,000 SEK
Prime Time License (Power Analysis, 1 year)	800,000 SEK

Table 6.4: Design tools cost for ASIC solution.

The hardware costs of ASIC scheme consists of an FPGA for verification and

the costs for taping out. The taping out cost is one-off investment. After taping out, the fabrication costs will be very cheap. The hardware costs are listed in Table 6.5.

Item	Price
FPGA for Verification (Kintex-7 Evaluation Kit)	20,000 SEK
Tape Out and re-spin	More than 1,000,000 SEK

Table 6.5: Hardware cost for ASIC solution.

Human resource is also a very important part of the total cost. ASIC development cycle is much longer than FPGA scheme, because the post-end design and verification is much more complex. A rough human resource requirement is listed in Table 6.6.

Design Flow	Time required
Hardware Logic Design	15 Weeks
Design Simulation & FPGA Verification	8 Weeks
Post-end Design	16 Weeks
ASIC Verification	30 Weeks
Documentation	8 Weeks

Table 6.6: Human resources required for ASIC solution.

The ASIC scheme requires a large start-up fund, takes longer to develop and the design is not flexible. But after the initial design is finished, the ASIC fabrication costs very little. Thus, ASIC scheme requires a large market to increase the quantity of sale, which can share the design costs to a large amount of products.

6.3.3 IP-core Scheme

IP-core is another scheme aimed at producing a processor. The only difference is that no real ASICs are fabricated by us. By purchasing the IP-core of digital cavity, the costumers can integrate it into their own ASICs. The digital cavity works as a single function of a much more complex chip. In work flow of this scheme, only logic design and verification are needed. IP-core is the cheapest scheme since there is no hardware product designed and manufactured. The only hardware cost is an FPGA used to verify the design. The software tools costs are listed in Table 6.7 and

the human resource required is listed in Table 6.8. In this scheme, simulation and verification is the most time consuming part. This can ensure the correctness of the design.

Software Tools	**Price**
IDE	Free
ISE Design Suit	Included in FPGA price
Debussy (Debug Tool)	Free
MatLab (Modelling)	5,000 SEK
ModelSim (Simulation Tool)	7,500 SEK

Table 6.7: Design tools cost for IP-core solution.

Design Flow	**Time required**
Hardware Logic Design	2 Weeks
Simulation & Verification	6 Weeks
Documentation	3 Weeks

Table 6.8: Human resources required for IP-core solution.

6.3.4 Conclusion

The three schemes have been discussed. They all have their own advantages and disadvantages. A comparison from the view of design cost, product cost, and potential market is shown in Table 6.9.

Scheme	**Design Cost**	**Cost of Each Product**	**Development Cycle Cycle**
FPGA	30,000 - 50,000 SEK	20,000 - 40,000 SEK	16 Weeks
ASIC	7,482,000 SEK	Lower than 100 SEK	77 Weeks
IP-core	12,500 SEK	0	11 Weeks

Table 6.9: Comparison of schemes.

From the Table 6.9, we see that the FPGA scheme has medium design cost, but the cost of each product is highest. The ASIC has a very high design cost, but the cost of each product is very low. IP-core scheme is the most easiest to implement, but its potential market and profit is limited.

So, we can draw a conclusion that FPGA is more suitable for a small volume manufacture, and the ASIC will have advantage when the quantity of sales is very large. The IP-core is most suitable when start-up fund is limited, or the situations that require the product to be highly flexible.

Appendix A

Deviation of the relation between the relative phase and cooling time

The cable exposes and contracts with temperature. If the change of temperature, ΔT is small, the change in length of the cable, ΔL, is given by:

$$\Delta L = -\alpha \Delta T L, \tag{A.1}$$

where L is the length of the cable and α is the linear expansion coefficient.

The relative phase ($\Delta\Phi$) which is the phase difference between the signals through the two cables, is proportional to the change of the wire. It is given by:

$$\Delta\Phi = -\frac{360^\circ}{\lambda_c}\alpha \Delta L + O, \tag{A.2}$$

where λ_c is the wavelength of the EM waves in the cable and the phase is measured in degrees and O is the offset.

Equation (A.1) substitute into Equation (A.2), Equation (A.3) are deducted that the change in the relative phase related to the change in the temperature:

$$\Delta\Phi = -\frac{360^\circ}{\lambda_c}\alpha \Delta T L + O. \tag{A.3}$$

The temperature, $T(t)$, of the cable as a function of Time, t. It is given by the Newton's law of cooling:

$$T(t) = T_a + (T_0 - T_a)exp(-kt), \tag{A.4}$$

where T_a is the temperature of ice (0 $^\circ C$), T_0 is the room temperature (21 $^\circ C$) and k is the cooling constant. Cooling or heating Time leads to the change in the relative phase. It is then given by:

$$\Delta\Phi = -\frac{360^\circ}{\lambda_c}\alpha\Delta L(T_0 - T(t)) = X_0 + X_1 exp(-kt), \tag{A.5}$$

where $X_1 = 360^\circ\alpha L(T_0 - T_a)/\lambda_c$ is a constant in the measurements.

Appendix B

Derivation of the relation between the relative amplitude and the cooling time

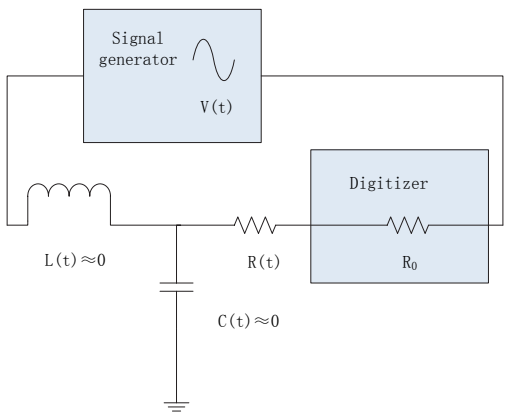


Figure B.1: The circuit model of phase measurement experiment

The circuit diagram equivalent to the setup is shown in Figure B.1. As the effect of inductance is minor in comparison of the effect of temperature change, besides the ground is far enough from the cable so the capacitance of the cable between

ground can be neglected. Use the following circuit equation:

$$I(t) = \frac{V(t)}{R_0 + R_t}, \tag{B.1}$$

where $I(t)$ is the current, $V(t)$ is the voltage of the signal generator (the AC power source), $R(t)$ is the resistance of the cable and R_0 is the impedance in the receiver of the oscilloscope. That the voltage drops V_0 across the resistance R_0 is

$$V_0(t) = V(t) - I(t)R(t). \tag{B.2}$$

Resistance of the cable, $R(t)$ at a particular temperature is given by:

$$R(t) = \frac{\rho(T(t)) \times L(T(t))}{S_0}, \tag{B.3}$$

where S_0 is the cross section of the cable and ρ is the electrical resistivity of the cable. The length as well the resistivity relate to the temperature. As the temperature change is not large, we use the temperature coefficient of resistivity, β and the linear approximation in the thermal expansion coefficient α:

$$L(t) = L_0[1 + \alpha(T - T_0)], \tag{B.4}$$

and

$$\rho(T) = \rho[1 + \beta(T - T_0)]. \tag{B.5}$$

During the cooling, $T(t)$ is given by Newton's law of cooling (Equation (A.4)). Using Equation (A.4), (B.3), (B.4) and (B.5), the resistance of the cable during the cooling can be deducted as a function of time:

$$\begin{aligned} R(t) &= \frac{\rho_0 L_0}{S_0}[1 - (\alpha + \beta)(T_0 - T_a)(1 - exp(-kt)) + \alpha\beta(T_0 - T_a)^2(1 - exp(-kt))^2] \\ &\approx \frac{\rho_0 L_0}{S_0}[1 - (\alpha + \beta)(T_0 - T_a)(1 - exp(-kt))]. \end{aligned} \tag{B.6}$$

As $\alpha \ll \beta$ for copper, we can further simplify Equation (B.6) to

$$R(t) = \frac{\rho_0 L_0}{S_0}[1 - \beta(T_0 - T_a)(1 - exp(-kt))]. \tag{B.7}$$

Using Equation (B.1), (B.2) and (B.7) we can get the expression for the voltage read by the oscilloscope:

$$V_0(t) = V(t) \times \frac{1}{1 + r_0[1 - \beta(T_0 - T_a)(1 - exp(-kt))]}, \tag{B.8}$$

where $r_0 = (\rho_0 L_)/(R_0 S_0)$.

Finally, we take the ratio of the voltage from the cooled cable to that of the voltage from the cable at the room temperature to decrease the errors in the measurement due to the fluctuations in the signal generator and the environment.

$$A(t) = \frac{C_0}{1 + r_0[1 - \beta(T_0 - T_a)(1 - exp(-kt))}. \tag{B.9}$$

Bibliography

[1] K. Karki, M. Torbjörnsson, J. R. Widom, A. H. Marcus, and T. Pullerits, "Digital cavities and their potential applications," *Journal of Instrumentation*, vol. 8, no. 05, p. T05005, 2013.

[2] O. Klein, S. Donovan, M. Dressel, and G. Grüner, "Microwave cavity perturbation technique: part i: principles," *International Journal of Infrared and Millimeter Waves*, vol. 14, no. 12, pp. 2423–2457, 1993.

[3] J. H. Scofield, "Frequency-domain description of a lock-in amplifier," *American Journal of Physics*, vol. 62, no. 2, pp. 129–132, 1994.

[4] F. Soukup, "Lock-in amplifiers," *Ceskoslovensky Casopis pro Fyziku Sekce*, vol. 28, pp. 271–274, 1978.

[5] S. Fu, A. Sakurai, L. Liu, F. Edman, T. Pullerits, V. Öwall, and K. Karki, "Generalized lock-in amplifier for precision measurement of high frequency signals," *Review of Scientific Instruments*, vol. 84, no. 11, 2013. [Online]. Available: http://dx.doi.org/10.1063/1.4827085

[6] J. E. Volder, "The cordic trigonometric computing technique," *IRE Transactions on Electronic Computers*, no. 3, pp. 330–334, 1959.

[7] V. Giannakopoulou and K. Masselos, "Hardware performance analysis of a parametric cordic ip," in *Signals and Electronic Systems (ICSES), 2012 International Conference on*. IEEE, 2012, pp. 1–6.

[8] H. Pahuja and P. S. Lavish Kansal, "Cordic algorithm implementation in fpga for computation of sine & cosine signals," *Int. J. Sci. Eng. Res*, vol. 2, pp. 1–6, 2011.

[9] A. Chrysafis, *Digital Sine-Wave Synthesis Using the DSP56001/2.* Application Note, Semiconductor Products Sector, Motorola, Inc., Austin, TX, 1988.

[10] P. Dienes, *The Taylor series: an introduction to the theory of functions of a complex variable.* New York, NY: Dover, 1957.

[11] LogicCore, *DDS Compiler v4. 0, IP.* Xilinx DS558, Mar, 2011.

[12] S. Gai-qing and D. You-er, "Application of lock-in amplifier in measurement of battery internal resistance," *Physics Examination and Testing*, vol. 2, p. 018, 2006.

LUND
UNIVERSITY

Series of Master's theses
Department of Electrical and Information Technology
LU/LTH-EIT 2014-416

http://www.eit.lth.se