The Design and Implementation of a New Wide-Range Frequency Detector

by

Steve Sunghwan Paik

Submitted to the Department of Electrical Engineering and Computer Science in partial fulfillment of the requirements for the degree of Master of Engineering in Electrical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

July 1998

© Steve Sunghwan Paik, MCMXCVIII. All rights reserved.

The author hereby grants to MIT permission to reproduce and distribute publicly paper and electronic copies of this thesis document in whole or in part, and to grant others the right to do so.

Author

Department of Electrical Engineering and Computer Science

July 28, 1998

Certified by

Lawrence M. DeVito
Engineering Manager, Analog Devices, Inc.
Thesis Supervisor

Certified by

James K. Roberge
Professor of Electrical Engineering, MIT
Thesis Supervisor

Accepted by

Arthur C. Smith
Chairman, Department Committee on Graduate Theses
The Design and Implementation of a New Wide-Range Frequency Detector

by

Steve Sunghwan Paik

Submitted to the Department of Electrical Engineering and Computer Science on July 28, 1998, in partial fulfillment of the requirements for the degree of Master of Engineering in Electrical Engineering

Abstract

In this thesis, I designed and implemented a wide range frequency detector for use in clock recovery and data retiming applications. The new detector works in conjunction with the existing “mid-range” frequency detector to accurately lock the VCO to the incoming data rate. The new detector consists of two halves: one to detect when the VCO is too fast, and one to detect when the VCO is too slow. The design and analysis of the new frequency detectors, in addition to a method for integrating it with the existing detector, is discussed. Simulation data of the new and original frequency detectors are used to support the concepts upon which the new detector is designed. Some topics for future work are suggested at the end of this thesis.

Thesis Supervisor: Lawrence M. DeVito
Title: Engineering Manager, Analog Devices, Inc.

Thesis Supervisor: James K. Roberge
Title: Professor of Electrical Engineering, MIT
Acknowledgments

Thanks to Larry DeVito for his patience and expertise in working with me to design and test a new wide range frequency detector. It was his guidance which made my work possible. Special thanks to Professor Roberge for his assistance in supervising my thesis for MIT and telling me that I drew my OR gates wrong.

I’d also like to thank the engineers at Analog Devices, especially Rosa Croughwell for her support in showing me what the existing frequency detector does, and Tony Freitas for helping me with the layout and parasitic extraction for my frequency detectors. It was with Rosa’s help that I was able to make sense of TC and ADICE.

Finally, I would like to thank Analog Devices, Inc. for participating in the VI-A internship program and creating an enjoyable experience for me there. This work could never have happened without ADI.
Contents

1 Introduction ........................................... 9
   1.1 Background Information ............................... 9
   1.2 A New Frequency Detector ............................ 10

2 Existing Frequency Detector ............................... 11
   2.1 Theory of Operation .................................. 11
       2.1.1 The Bicycle Wheel Analogy ...................... 11
       2.1.2 Operation of the Existing Detector ............... 12
   2.2 Performance of the Existing Detector ................. 14
       2.2.1 Simulation Data .................................. 14
       2.2.2 Observations in Lab ............................. 17
   2.3 Limitations of the Existing Detector ................. 18

3 Design of the New Frequency Detector ..................... 20
   3.1 The VCO-Slow Detector ................................ 21
       3.1.1 Design of the VCO-Slow Detector ................ 23
       3.1.2 Simulation of the VCO-Slow Detector .............. 26
   3.2 The VCO-Fast Detector ................................ 28
       3.2.1 Design of the VCO-Fast Detector ................. 29
       3.2.2 The Next Revision ............................... 31
       3.2.3 Simulation of the VCO-Fast Detector .............. 34
   3.3 Integrating Both Detectors ........................... 39

4 Integration of the New Frequency Detector .................. 40
   4.1 Original VCO Setup .................................. 40
4.2 Integrating the Original and New Detectors ............... 41
  4.2.1 Simulation of the Two Detectors .................... 41
  4.2.2 Combining the Outputs ............................... 45

5 Conclusions ................................................. 47
  5.1 Problems with the New Frequency Detector ............... 47
    5.1.1 Performance of the VCO-Slow Detector ............. 47
    5.1.2 Performance of the VCO-Fast Detector ............. 47
  5.2 Alternative Methods of Frequency Detection .............. 48
  5.3 Final Conclusions ...................................... 49

A Figures ................................................... 50
List of Figures

2-1 VCO is faster than data................................................. 13
2-2 VCO is slower than data................................................ 13
2-3 Original detector simulated with MAX data pattern over 500ns....... 15
2-4 Original detector simulated with 128-bit PRBS data pattern over 500ns... 16
2-5 Original detector simulated with 15% bimodal jitter on 128-bit PRBS data pattern over 500ns............................. 17
2-6 Data can look both fast and slow........................................ 18

3-1 Simple VCO-Slow detector................................................ 21
3-2 The VCO-slow detector is disabled half the time....................... 22
3-3 An edge-detector circuit to trigger flops on both the rising and falling edges of the input data stream........................................... 23
3-4 Parallel implementation of the VCO-slow detector....................... 24
3-5 Aperture distortion in the latch caused by the RESET signal.............. 25
3-6 VCO-Slow detector simulated with MAX data pattern over 500ns........... 26
3-7 VCO-Slow detector simulated with 128-bit PRBS data pattern over 500ns... 27
3-8 VCO-Slow detector simulated with 128-bit PRBS data pattern with 15% bimodal jitter over 500ns........................................... 28
3-9 3 clock edges between data transitions indicate VCO is fast............... 29
3-10 Original concept for VCO-Fast detector..................................... 30
3-11 Up to a 50% error can go unnoticed through the detector shown in Figure 3-10................................................................. 31
3-12 There is a finite propagation delay in the AND gate of the circuit shown in Figure 3-10................................................................. 32
3-13 The AND gate never goes high, because of the propagation delay in the circuit shown in Figure 3-10. .......................... 33
3-14 Up to a 25% error can go unnoticed when using a parallel combination of the circuit shown in Figure 3-10. .......................... 34
3-15 The new front end for the VCO fast detector. .......................... 35
3-16 An error as low as -20% will trip the detector shown in Figure 3-15. .......................... 36
3-17 VCO-Fast detector front end simulated with MAX data pattern over 500ns. .......................... 37
3-18 VCO-Fast detector front end simulated with 128-bit PRBS data pattern over 500ns. .......................... 38
3-19 VCO-Fast detector front end simulated using 128-bit PRBS with 15% bimodal jitter data pattern over 500ns. .......................... 39

4-1 Block diagram of the frequency detector, charge pump, and VCO. .......................... 41
4-2 Simulation of both detectors using a MAX data pattern. .......................... 42
4-3 Simulation of both detectors using a 128-bit PRBS data pattern. .......................... 43
4-4 A closer look at the 128-bit PRBS data pattern simulation. .......................... 44
4-5 Simulation of both detectors using 15% bimodal jitter in conjunction with a 128-bit PRBS data pattern. .......................... 44
4-6 Modified block diagram for PLL incorporating the new frequency detector. .......................... 45

A-1 The VCO-Slow detector. .......................... 51
A-2 The VCO-Fast detector. .......................... 52
A-3 The VCO.SLOWDETV2.1 block. It is the major building block in the VCO-Slow detector. .......................... 53
A-4 The XFAST.LATCHWLVLSHFT block. It is a latch with a level shifted output. .......................... 54
A-5 The OR4INLOW block. It is a simple 4 input OR gate. .......................... 55
A-6 The OR4IN block, a five input OR gate. .......................... 56
A-7 The XFAST.FLOP block. It is the standard flip-flop used in the VCO-Fast detector. .......................... 57
A-8 The XFAST.LATCH block. The basic latch used in the XFAST.FLOP. .......................... 58
A-9 The XFAST.FLOPWAND block. It is a flip-flop combined with an AND gate as its input. .......................... 59
A-10 The XFAST_LATCHWAND block, used in the XFAST_FLOPWAND. . . . 60
A-11 The XFAST_FLOPWCLR block. It is a flip-flop with a CLEAR function. . 61
A-12 The XFAST_LATCHWCLR block, used in the XFAST_FLOPWCLR. . . . 62
A-13 The XFMAND block, a 2 input AND gate. . . . . . . . . . . . . . . . 63
A-14 A multiple input AND gate. . . . . . . . . . . . . . . . . . . . . . . . . 64
Chapter 1

Introduction

1.1 Background Information

Clock recovery and data retiming has traditionally been done using PLLs referenced to a crystal oscillator or with the use of a SAW filter. These methods are generally large and expensive, making the alternative of a single clock recovery and data retiming IC more attractive. The problem with using a single monolithic IC PLL is that the original clock rate of random data is difficult to determine without a fixed frequency reference.

Analog Devices currently produces a monolithic clock recovery and data retiming IC (AD808) which adheres to the SONET (Synchronous Optical NETwork) specifications. SONET allows data streams of different formats to be combined onto a single high-speed fiber optic synchronous data stream. The AD808 uses a frequency detector which has a range of ±25% of the center frequency of the on-chip VCO. Thus, the VCO's range must be limited and its center frequency must be laser trimmed and tested on each part before it is packaged and sold. This is done at a tremendous cost, and is available only on the bipolar process which Analog Devices is currently using. Although this solution works, Analog Devices would like to produce the same circuit on a CMOS process. Laser trimming of components is not an option on CMOS, and the on-chip VCO center frequency can drift by 50%. This requires that the VCO be designed with a ±50% range, which necessitates a new wide-range frequency detector to help lock the PLL. To my knowledge, implementing such a wide-range frequency detector of this type onto an IC has never been done before.
1.2 A New Frequency Detector

This thesis outlines the design and implementation of a new, wide-range frequency detector. It is not necessary for the new detector to replace the original one, if the two detectors are capable of coexisting and working together. However, the new detector should have at least a ±50% error detection range.

Chapter 2 discusses the operation of the existing frequency detector, looking at its capabilities and limitations. Simulation results, using the original detector and sweeping the clock frequency, are presented here.

Chapter 3 shows the development of the new frequency detectors in their final form. Two separate detectors are developed, one for the case when the VCO is too fast, and one for when the VCO is too slow. The design and simulation of these detectors is covered here.

Chapter 4 addresses the issue of combining the new wide range frequency detector with the original one. It is found that the two detectors will work together with the addition of an OR gate to combine their signals.

Chapter 5 is the conclusion of this thesis, and investigates some topics that can be worked on in the future to improve upon the current design.
Chapter 2

Existing Frequency Detector

2.1 Theory of Operation

The frequency detector used in the AD808 is based on the rotational frequency detector originally developed for Analog Devices by John Newton. This detector uses data transitions to sample the different phases of the VCO. If there is a frequency error, there will be a change of phase (in the VCO) in time which is proportional to the error. The larger the frequency error, the faster the phase of the VCO will change.

2.1.1 The Bicycle Wheel Analogy

A common analogy for this frequency detector is a bicycle wheel with a ribbon tied around one of its spokes in a dark room. The ribbon will represent the phase of our VCO. Now, imagine a strobe light being triggered in the same room. The strobe light represents data transitions, and serves to “sample” phases of the VCO. If the frequency of the strobe light is the same frequency as the bicycle wheel’s rotational rate, the ribbon will appear in a static position to the observer.

If the frequency of the strobe light is faster or slower than the rotational rate of the wheel, the ribbon will begin to move either clockwise or counterclockwise, depending upon the nature of the frequency error. For example, let’s assume the wheel is rotating clockwise with respect to the viewer. If the frequency of the strobe light is slower than the rotational frequency of the wheel, the ribbon will begin to move clockwise because the wheel completes a little more than one revolution between each strobe flash. Likewise, if the strobe light
is slightly faster than the wheel, the ribbon will begin to move counter-clockwise. But, if the strobe light is too fast, the ribbon will eventually reverse direction and move clockwise. When we make the wrong conclusion about the direction of the frequency error, we will refer to it as aliasing. This happens when either the strobe light is more then twice as fast as the wheel, or less than 2/3 as slow.

At this point, we have created the analogy using a maximum data rate. Now, let's insert missing edges in the strobe light; sometimes we get a flash, and sometimes we get a missing edge and the room remains dark. This complicates our detection scheme, because the wheel can appear to go in both directions for both types of frequency errors (fast and slow). For example, if the strobe frequency is faster than the rotational frequency, the ribbon should be moving counter-clockwise. Now, let's assume we're missing three or four edges, so that four “fast” data edges will look the same as three “slow” data edges. The ribbon will have moved clockwise, although the frequency error dictates that it should have moved counter clockwise. Thus, the missing edges produce a form of aliasing which can lead the detector to conclude the wrong frequency error exists. This effect makes the current form of detection unreliable for large frequency errors.

2.1.2 Operation of the Existing Detector

In the AD808, the VCO generates a quadrature output. This output allows one to derive four distinct phases of the clock (A,B,C, and D) by using various combinations of the quadrature signals. The phases of the clock can be thought of as quadrants of a circle, with a data phasor which spins around the clock. The data is ideally locked to the rising edge on the A phase of the clock. Thus, the B-C boundary is used to detect frequency errors because it is farthest away from the locked data.

If the clock is faster than the data, eventually the data phasor will exhibit a B to C transition, as shown in Figure 2-1. If the clock is slower than the data, we will observe the opposite, a C to B transition (shown in Figure 2-2). If the clock is locked to the data, we will observe hits in the A and D quadrants, dependent upon jitter on the data and phase noise in the VCO. Thus, when the VCO is properly locked to the data, we should not observe any hits in the B or C quadrants, much less a transition between the two quadrants which indicates the existence of a frequency error.

This concept works well for small frequency errors, during which the clock and data
Figure 2-1: VCO is faster than data.

Figure 2-2: VCO is slower than data.
drift slowly in phase. For example, if we have max data coming in, and the clock is a little bit faster than the data, we will observe many hits in A, followed by many hits in B, and so on. The phase of the clock will "walk" around the circle.

If the frequency error is a little larger, perhaps we will see a couple hits in A, a couple hits in B, and the phase of the clock begins to "walk" faster around the circle. Now, if we have some missing data edges, we no longer see an A to B transition, but perhaps an A to C transition (depending upon the number of data edges that are missing). Thus, we can skip over quadrants when there is a modest frequency error with some missing edges. A frequency error of over 25% has the ability to skip quadrants at a max data rate, and when missing edges are encountered, smaller frequency errors can do the same.

2.2 Performance of the Existing Detector

Since the existing frequency detector has been designed and implemented on several products built by Analog Devices, its performance has been well characterized. Of particular importance are the detection ranges and jitter tolerance.

2.2.1 Simulation Data

In simulations, the existing frequency detector performs reliably for frequency errors less than ±25%. Several simulations were done to measure the performance of different aspects of the frequency detector. All simulations were done over a 500ns simulation time using a 1.608ns data period and sweeping the clock period in increments of 0.05ns. This corresponds approximately to increments of 3% in frequency error between the clock and the data. The 1.608ns data rate corresponds to 622MHz, which is the operating frequency of the AD808.

Frequency error is defined as \( \frac{\text{Period}_{\text{Data}} - \text{Period}_{\text{VCO}}}{\text{Period}_{\text{VCO}}} \). The original detector produces two different signals, FUP and FDN. A FUP pulse occurs on a C to B transition, indicating that the VCO is slow. Thus, an FUP pulse will work to increase the VCO frequency. The FDN pulse is just the opposite, occurring on a B to C transition. It indicates that the VCO is too fast, and it will lower the clock frequency. The more hits registered from the detector, the larger the frequency error. The data plotted in the following figures is the output of the FUP and FDN signals from the original detector, versus the frequency error. In the AD808, the FUP and FDN signals go into a charge pump circuit, which will raise or lower
the frequency output from the VCO.

MAX Data Pattern

The results of the first simulation are shown in Figure 2-3. This simulation used a MAX data pattern to show the general functionality of the detector. In this simulation, notice that false FDN hits occur past -30% frequency error, but there are no false FUP hits at all. False hits are a big problem, and it is reassuring to see that for a MAX data pattern, there are no false hits within a ±30% frequency error range, which is the effective range of this frequency detector. False hits are not fatal if there are more correct hits than false hits, and this is true for this simulation except for large, negative frequency errors.

128-Bit PRBS Data Pattern

The next simulation used a 128-bit PRBS (Pseudo Random Bit Stream) data input to determine how well the circuit responds to missing edges. The results of this simulation are shown in Figure 2-4. The maximum number of hits for both the FUP and FDN detectors have gone down by a factor of four, indicating that the detector has become less sensitive due to the missing edges. However, the number of hits seen within ±20% frequency error is
the same as in the MAX data simulation. This shows that the original frequency detector is still sensitive to cycle slips in the PRBS data pattern for small frequency errors, but once the frequency error becomes larger, the detector will not see it. Another feature to notice is that false hits begin to occur at ±10% frequency error. This is undesired, but not fatal as long as there are more correct hits than false hits.

The charge pump does not exactly cancel an FUP hit with an FDN hit; one signal may be slightly larger than the other. Thus, false hits do not exactly cancel out correct hits. Ideally the detector function if there is one more correct hit than false hit. However, because the FUP and FDN pulses do not cancel each other out, it is possible that a smaller number of false hits can dominate over a larger number of correct hits. Therefore, although false hits can be tolerated in this detector, they are undesirable. This simulation shows that the detector is still reliable to approximately ±35% frequency error.

15% Bimodal Jitter With 128-Bit PRBS Data Pattern

The third simulation adds 15% bimodal jitter to the 128-bit PRBS data input. Jitter and phase noise are found in some amount in many fiberoptic transmitters, requiring the frequency detector to be robust enough to work with some jitter. Immunity to bimodal
Figure 2-5: Original detector simulated with 15% bimodal jitter on 128-bit PRBS data pattern over 500ns.

jitter is a particularly useful feature because the dispersion effects of a fiberoptic cable usually shift the data edges either early or late, creating bimodal jitter.

Figure 2-5 shows the simulation results to the bimodal jitter input. The performance of the original detector has become considerably worse, showing false hits at all frequency errors, and being reliable to ±25% frequency error. Although the circuit is tolerant of bimodal jitter, its performance has degraded significantly from the previous situation with just random data. However, for small frequency errors, the detector exhibits the same amount of correct hits as seen in the previous two simulations. This is important, because it shows that the frequency detector is still able to detect cycle slips and determine the correct frequency error.

2.2.2 Observations in Lab

The AD808 is an existing product which has already been manufactured. Its performance has been well tested and measured both in lab and in the field. The simulation results correspond nicely with the observed performance of the actual circuit in lab. Of notable importance is the susceptibility of the frequency detector to bimodal jitter. In the past,
this has been the major weakness in the detector, and it still hinders performance.

For the bimodal simulation, observe that false hits occur at zero error. This, combined with unequal FUP and FDN signal strengths, can cause false locking of the detector for small frequency errors. Thus, although there are more correct hits than false hits near zero frequency error in the plot, the frequency detector is susceptible to false locking.

2.3 Limitations of the Existing Detector

From the simulated and observed laboratory data, it is obvious that the original detector could be used to detect at most a $\pm 25\%$ error between the data and the clock. This is not even close to the desired error detection range of $\pm 50\%$ or more.

The problem with the existing frequency detector is that it does not keep track of time. Although this is not important when acquiring to a max data pattern, a random data pattern with missing edges has the ability to confuse the detector. For example, the existing frequency detector can receive a hit in B, and the data phasor can spin around the circle a couple of times, before the detector receives another hit. In this way, two (or more) bits of data can register as either the VCO being too fast, or too slow, and the detector has no way to distinguish between the two as shown in Figure 2-6. This is the aliasing problem, making the detector ineffective for large frequency errors, over $\pm 25\%$. 
The two problems with this detector are its limited detection range, and its susceptibility to bimodal jitter. Bimodal jitter has the effect of shifting some of the data edges and making them fall either early or late. This can be a problem as we walk around the circle, because we can get into a case where there are several B-C or C-B transitions in a row. Because bimodal jitter is commonly found in applications where this chip will be used, it is important that the new frequency detector be immune to it.
Chapter 3

Design of the New Frequency Detector

As discussed in the previous chapter, the existing frequency detector has a limited operating range. This is a result of fact that the data is sampling the phases of the clock. Aliasing occurs because the data has missing edges and the detector is not keeping track of time. Thus, many bits may pass between data edges and the frequency detector will not have taken this into account.

There are several important features which must be incorporated in the design of the new frequency detector:

1. It should either replace or co-exist with the existing one.

2. It cannot give any false hits.

3. It must be reliable to detect at least a ±50% frequency error.

To create a new, wide-range frequency detector, it is important to account for the amount of time that passes between data edges, as well as sampling the clock phase with each data edge. The challenge of designing the new frequency detector can be split into two separate parts:

1. Detecting when the VCO is faster than the data rate.

2. Detecting when the VCO is slower than the data rate.
It is easier to look at half of the problem at a time, and understand what needs to be done in order to solve it, rather than trying to solve all of it at once. This chapter will explain the theory and concepts upon which the new frequency detectors are designed.

3.1 The VCO-Slow Detector

Detecting when the VCO is slower than the data rate is straightforward. When the VCO is slower than the data, it is conceivable that two data edges will occur within one full clock period. This is impossible when the VCO is correctly locked to the data, so it is correct to assume that when one sees two data edges within one clock period, the VCO is slow.

Two data edges within one clock period would correspond to a one-bit wide piece of data (unless the clock was really slow, in which case it could be a two or more bit wide piece of data). This method of detection is somewhat data dependent in that it looks for a one-bit wide piece of data, but it assumes that this is a reasonable event to expect in a random data stream. Thus, the simplest implementation of the VCO-Slow detector is two resettable flip-flops, as shown in Figure 3-1.

Although the VCO-Slow Detector is a simple circuit, it is difficult to implement an edge triggered reset within a flop. It is, however, easy to implement a reset when the clock remains high or low, so a parallel architecture can be used to achieve the same effect.

The flops in this circuit are reset when the RESET signal is high, so it is necessary to use \( \frac{\text{clock}}{2} \) such that the transitions in the RESET signal occur along the B-C boundary.
Figure 3-2: The VCO-slow detector is disabled half the time.

This is done so that a correctly locked VCO which will receive data hits on the rising edge of the clock has less chance of declaring a false hit due to bimodal jitter or phase noise. The latches are reset every other clock period to prevent the aliasing problem discussed earlier. This ensures that when two data edges are seen, it has happened within one full clock period. Although this will work, there is a problem with this implementation. The detector is disabled half of the time as seen in Figure 3-2, so it is not very sensitive. Adding to this is the fact that the first flop triggers on the rising edge, and the second one triggers on the falling edge, making the detector sensitive to the particular order in which the data edges arrive. It would be nice to have flip-flops that trigger on both edges, and reset on clock edges, but neither of these features were found. Both of these problems can be (and are) fixed with a parallel architecture.

One way to implement an edge-triggering feature is to use an edge-detector as shown in Figure 3-3. This idea was explored, but it was found that the delay between the reset of the flip-flop, and its ability to sample on a data edge immediately following a clock edge was severely hindered. The edge-detector would assert the RESET high for approximately 200ps, disabling the VCO-Slow detector for over 12% of the data period. As one can see from the timing diagram shown in Figure 3-3, if the edge-detector asserts the RESET signal for any amount of time, that time will directly shorten the operative time of the VCO-Slow detector (the time that the RESET signal is low).

There are many ways to implement the VCO-Slow detector. The key concept to keep in mind is that two data edges cannot exist in a full period of the clock when their frequencies are equal. However, this becomes complicated if either the clock or the data has jitter on it. In this case, it is possible that two data edges will be seen during a full clock period, so
Figure 3-3: An edge-detector circuit to trigger flops on both the rising and falling edges of the input data stream.

it is recommended to use the B-C boundary for this detection scheme.

3.1.1 Design of the VCO-Slow Detector

The first step in designing the VCO-slow detector was to verify that the ideal topology works. This was done using ideal edge triggered, edge-resettable flip-flops working with ideal gates that have no delays. The topology for the VCO-slow detector was verified as being feasible. The next step was to design a chip-level circuit with real delays and parasitics, and show that it can work.

Although the simple ideal VCO-slow detector shown in Figure 3-1 does indeed work, it was found to be very insensitive. To increase its sensitivity, two levels of parallelism are necessary. One to take care of the reset signal problem, and one to take care of the fact that the flip-flops can only be triggered in one direction (either on the rising edge, or the falling edge). The parallel architecture for this circuit is shown in Figure 3-4. As discussed earlier, edge-detectors cannot be used on the flops because of the timing involved in resetting the flops every clock period and using the data to clock the flops. The parallelism allows the circuit to begin detecting as quickly as possible on the edges of \( \frac{Clock}{2} \), avoiding the delay due to the pulse width associated with the edge-detector.

This circuit was designed using Analog Devices’ XF 1.5 process. Simulations on this circuit were run with a clock period between 1ns and 3ns, and a data input of 622MHz. The simulations show that the detector works as expected. The Freq.Up pulse begins to activate around 15% frequency error, but there are not many of them. The reason that the
Figure 3-4: Parallel implementation of the VCO-slow detector.
Fig. 3-5: Aperture distortion in the latch caused by the RESET signal

Freq. Up pulse activates around 15% frequency error is due to duty cycle distortion in the activity of the RESET signal. The problem here is in the implementation of the RESET function in the latches. The particular latch used in making the resettable flip-flops is shown in Figure A-12, found in the appendix. RESET is asserted when the CLR_LL signal goes high, which will cause the output to go low. When RESET is released, the transistor Q13 requires some time to build up current, which creates a transient in the amount of time it takes for the latch to become active. The circuit spends more time being inactive than being active. This can be thought of as "aperture distortion". Figure 3-5 shows the duty cycle distortion which causes the latch to be inactive for a longer period of time than it is active.

Because it is desirable to increase the detector’s sensitivity, more parallelism was added to allow the circuit to work on all four boundaries of the clock, not just the B-C boundary. This was done by dividing the other quadrature clock signals by two. Now, the circuit will work on the A-B, B-C, C-D, and D-A boundaries. The circuit shown in Figure 3-4 is copied four times to achieve this parallelism. This makes the detector four times as sensitive and produces many more hits than the original implementation.

Using the D-A boundary may be undesirable because when the data is properly locked to the clock, there is a lot of activity in that area. Although this is a concern, the simulations have not shown any false hits for small frequency errors, so using this boundary appears to be safe, as long as the phase jitter of either the clock or the data stream is reasonable. As long as no false hits are produced, the detector topology is acceptable. If, in later processes, false hits become a problem, this detector can be desensitized by removing some, or all of
Figure 3-6: VCO-Slow detector simulated with MAX data pattern over 500ns.

the parallelism. It is possible to remove just the D-A detector, or even the A-B and C-D detectors as well.

3.1.2 Simulation of the VCO-Slow Detector

The VCO-Slow detector (the parallel detector, using four copies of Figure 3-4 on the four boundaries of the clock) was simulated under the same conditions as the original detector in Chapter 2 (data rate of 622MHz, 500ns simulation time). The simulations show no false hits and that detector performance improves as the frequency error gets larger.

MAX Data Pattern

The MAX data pattern is important because it shows the minimum frequency error required to activate the detector. The simulation data is shown in Figure 3-6. For this data pattern, the VCO-Slow detector begins to register hits around 16% frequency error. The 16% figure is a result of the aperture reduction discussed earlier, and is not predicted from the topology of the circuit itself. As the devices and data rate change, the actual frequency error at which the circuit begins to activate is likely to change as well. This is well within the range of the original frequency detector, so the new VCO-Slow detector will work with it.
Figure 3-7: VCO-Slow detector simulated with 128-bit PRBS data pattern over 500ns.

The plot of the detector begins to flatten out for large frequency errors. Although this may appear to indicate that the detector is less sensitive at higher frequency errors, the opposite is actually true. For very large frequency errors, the number of hits that are possible over a 500ns time period is less than the maximum number of hits possible for smaller frequency errors. This is happening because the clock period becomes longer for larger frequency errors. For example, at a 50% frequency error, the clock period is 3.2ns, and for a 10% frequency error, the clock period is 1.8ns. Thus, at 50% error, there is a maximum of 156 possible hits (over a 500ns simulation time) while at 10% frequency error, there can be as many as 277 hits.

**128-Bit PRBS Data Pattern**

The effects of PRBS data is seen in the data for this simulation, plotted in Figure 3-7. There are less than half as many hits, due to the fact that not all the data in the PRBS is a single bit wide.

**15% Bimodal Jitter with 128-Bit PRBS Data Pattern**

The results of this simulation are shown in Figure 3-8. The detector appears to have become
more sensitive to the frequency error. This is because the bimodal jitter tends to make a one-bit wide piece of data look even shorter, which is more likely to be detected in the VCO-Slow detector.

The problem here is that the original 16% frequency shift shown for the MAX data case is highly dependent on device parameters. The simulations were done at room temperature with nominal models. They were not repeated to determine the robustness of the circuit over process and environmental variations. These simulations were done to prove that the idea of wide range frequency detection is feasible. It is possible for bimodal jitter to confuse this detector and produce false hits, similar to the case of the original frequency detector.

3.2 The VCO-Fast Detector

Detecting when the VCO is faster than the data rate is more difficult than detecting that the VCO is slower. This is because of the missing edges in the data stream. The VCO-Slow detector takes advantage of the fact that there is a minimum time during which there should be at most only one data edge. If there is more than one data edge in that minimum amount of time, then it is a fact that the VCO is slow. On the flip side, there is no maximum time
Figure 3-9: 3 clock edges between data transitions indicate VCO is fast.

during which there must be at least one data edge. The data stream can have many missing edges, and there is no guarantee that an edge will appear within any amount of time.

To create a VCO-Fast detector, some assumptions need to be made about what kind of data the detector will see. The key here is that data edges occur at random; it is not guaranteed that two consecutive edges represent one bit, two bits, etc. It is desirable to make as few assumptions about the data as possible, making sure that the circuit is not overly dependent on statistics such as data density and data patterns. The assumption that is made for this circuit is that there will be some consecutive data edges that are one bit wide within the incoming data stream.

As long as this detector sees a one-bit wide piece of data within a certain number of data edges, it can detect the frequency error.

3.2.1 Design of the VCO-Fast Detector

The VCO-Fast detector has gone through several revisions. The two major ones will be discussed here. The first concept for the detector was to count the edges of the clock. If three or more edges are seen between each data transition, then the detector knows that the clock is fast. Figure 3-9 shows the basic concept for this detector. The output of this detector goes high when it counts three or more clock edges between data edges. This output is put into a series of flops that keep track of the previous state. If all of the flops indicate that the VCO is always faster than the data, then the Freq_Dn output will be activated. This concept was implemented using the circuit in Figure 3-10.

Another copy of the circuit needs to be used in parallel with this one, such that the data only keeps one set of flops reset at a time in the front end of the detector. There only needs to be one series of flops for the whole circuit, so the front end can be paralleled to provide the desired sensitivity.
This circuit works well if all the elements are ideal, but there are several problems with it. The first problem is that there is a wide range of bits that will “slip” through the detector, even though the VCO is fast. Although a 0% error can be detected, it is possible for a 50% error to go unnoticed. This effect is illustrated in Figure 3-11. Next, there are some problems with the propagation delays of the circuit elements. Of particular importance is the \( t_{pd} \) of the AND gate. This delay desensitizes the detector because even if three clock edges have been detected, if a data edge comes too soon (before the information had propagated through the AND gate) the front end flops will reset and the frequency error will not be noticed. Figure 3-12 shows a typical timing diagram of the front end of the VCO-Fast detector shown in Figure 3-10. The output of the flops are shown, and it is seen that once the last flop (Q4) goes high, the output of the AND gate goes high approximately 200ps later. The timing diagram is not done to scale; it is slightly exaggerated to show the propagation delay in the AND gate and its affect on the circuit. In Figure 3-13, the AND gate never goes high because a data edge came during the time the data was propagating through the AND gate. The propagation delay of the AND gate is approximately 200ps, which causes at least a 12% error in the front end of the detector. Because of this delay, the architecture of the VCO-Fast detector had to be redone to compensate for it.

The first idea for fixing the architecture is to add more parallelism, using both the I and Q phases of the VCO to detect for a fast clock. This lowers the maximum amount of error that can go unnoticed to 25% (shown in Figure 3-14, but it does not fix the errors due to
the propagation delays in the circuit. The ideal architecture does work, but a few changes need to be made to allow it to be easily integrated into an IC.

3.2.2 The Next Revision

As shown in the previous section, using three clock edges for the VCO Fast Detector yields an ideal probability of detection of 100% for a 25% frequency error. This does not include parasitics or gate delays in the calculation, so that number will be higher. Thus, the front end needs to be redesigned to detect smaller frequency errors.

The new idea is to use a combination of the I and Q clock edges, and count for only four of them. If the detector sees four clock edges between two data edges, it will note that the clock is fast. If this happens for many pieces of data, the detector will signal that the clock is, indeed, fast. The new VCO-Fast detector circuit is shown in Figure 3-15. The circuit has a chain of flops that is six flops long, allowing the circuit to keep track of the previous twelve data transitions. Figure 3-16 shows that for the new front-end, a -25% frequency error can be detected, and anything greater than 0% error will be detected. This would appear to be too sensitive and will give false hits.

Note that the AND gate in the front-end of the new detector (shown in Figure 3-15) connected to the first column of four flip-flops, will add some propagation delay. This,
Figure 3-12: There is a finite propagation delay in the AND gate of the circuit shown in Figure 3-10.
Figure 3-13: The AND gate never goes high, because of the propagation delay in the circuit shown in Figure 3-10.
Figure 3-14: Up to a 25% error can go unnoticed when using a parallel combination of the circuit shown in Figure 3-10.

added to the parasitics, will sufficiently desensitize the detector to keep it from registering false hits. Now, it is seen that if there is no propagation delay in the AND gate, the detector would be too fast. Of course, if there is too much delay in the AND, then the detector would be too slow. Having the circuit be dependent on the propagation delay of a logic gate is difficult to work with, because that delay cannot be well controlled from part to part. One method to remove the propagation delay of the AND gate could be to use latches to hold the value of flops Q3 and Q4, so that the AND gate has plenty of time to settle to the correct value, and this can be read. This is something that should be considered for future development of this circuit. The second AND gate, however, is between two flip-flops so the timing issues related to the first AND gate do not apply to it. Refer to Figure A-2 in the appendix to see the actual implementation of the VCO-Fast detector.

3.2.3 Simulation of the VCO-Fast Detector

The VCO-Fast detector (shown in Figure 3-15 and parallelized to alternate which front-end is active) was also simulated under the same conditions as the original detector in Chapter 2, and the VCO-Slow detector discussed earlier. The data gathered from testing the VCO-Fast detector is different from the data gathered from the VCO-Slow detector. For the
Figure 3-15: The new front end for the VCO fast detector.
Figure 3-16: An error as low as -20% will trip the detector shown in Figure 3-15.

VCO-Fast detector, the output of the front end (which is connected to the series of flops that keep track of the previous state) is plotted instead of the actual output of the detector. This is done to give more resolution of the activity at the front end of the detector.

**MAX Data Pattern**

The MAX data pattern is important in testing the characteristics for the VCO-Fast detector. It determines the minimum frequency error that will be detected when a single-bit wide piece of data is present. The MAX data pattern will also determine how much frequency error will “saturate” the detector so that the detector will always indicate a frequency error.

The data is plotted in Figure 3-17. From the plot, the minimum frequency error to be detected by this detector is at 15% frequency error. It is important to note that there are no false hits due to the MAX data pattern. From this, we can conclude that as long as there is a one-bit wide piece of data which has been sampled within the series of flops, there will be no false hits from the detector.

The maximum frequency error that can pass through this detector is more difficult to determine from the plot because of the fact that there is a maximum number of hits possible per simulation. In the VCO-Slow detector, the maximum number of hits per simulation
Figure 3-17: VCO-Fast detector front end simulated with MAX data pattern over 500ns.

time was dependent upon the clock frequency because the flops are reset by the clock. In the VCO-Fast detector, the opposite is true. The maximum number of hits per simulation is determined by the data pattern, because it is the data which is resetting the flops and sampling the phases of the clock. Thus, the maximum number of hits is independent of clock frequency, and only depends on the data frequency and data density. For this particular simulation, the maximum number of hits was 155, and that was achieved at 50% frequency error.

**128-Bit PRBS Data Pattern**

Using a 128-bit PRBS data pattern, it is no surprise that there are many “false” hits coming out of the front end of the VCO-fast detector, shown in Figure 3-18. This is expected, and shows that the detector is more sensitive due to the smaller number of data edges. There are less hits overall because the width of a hit is determined by the data pattern, and in this case, because there are missing edges, some of the hits are longer than others.

The “false” hits can be used as a baseline, and it can be seen that the detector starts to see other data at a frequency error of approximately 11%. The detector also saturates at approximately 45% frequency error, which is consistent with the results from the MAX
Figure 3-18: VCO-Fast detector front end simulated with 128-bit PRBS data pattern over 500ns.

data simulation. These “false” hits are a result of the longer pieces of data (two-bit wide or more) coming through the front-end of the detector. It is important to understand that these false hits will not be seen at the output of the VCO-Fast detector, because the chain of flops will act to “filter” them out.

15% Bimodal Jitter With 128-Bit PRBS Data Pattern

The data for this simulation is plotted in Figure 3-19. Comparing it to the previous plot, it is shown that bimodal jitter is detrimental to the VCO-fast detector. The detector is capable of detecting a reasonable amount of frequency error, but it does not saturate until there is approximately an 80% frequency error.

The simulations show that the VCO-Fast detector begins to activate near 15% frequency error. They have also determined that the detector has more delay in the front end than originally thought, so there is no concern for the possibility of false hits due to the fact that it is only using four edges of the quadrature clock.
Figure 3-19: VCO-Fast detector front end simulated using 128-bit PRBS with 15% bimodal jitter data pattern over 500ns.

3.3 Integrating Both Detectors

The new frequency detectors are reliable only for errors greater than ±20%. This makes the existing frequency detector necessary for mid-range tuning of the VCO. It is critical that the new frequency detectors be able to work with the original frequency detector. The next chapter will describe how the two detectors are integrated and show simulation data with the two detectors working together.
Chapter 4

Integration of the New Frequency Detector

The design of the new frequency detector was discussed in the previous chapter. It was shown that the new detector can reliably detect frequency errors as small as ±20%. The new detector is not accurate enough to work with the phase detector to lock the VCO to the incoming data stream. The original detector is still necessary to tune the VCO for small frequency errors, and the new detector takes over for large errors.

When combining the two detectors, it is important that they do not interfere with each other. It has already been shown that false hits occur in the original detector for large frequency errors; the new detector needs to be integrated in such a way that the false hits will not interfere with it.

4.1 Original VCO Setup

In the AD808, the original frequency detector is connected directly to a charge pump. The charge pump controls the input voltage going into the VCO. A simple block diagram of this setup is shown in Figure 4-1. The FUP and FDN outputs of the frequency detector move the VCO frequency up and down, respectively. The outputs from the frequency detector are clocked from the VCO, so each pulse of FUP or FDN will be exactly one clock period wide.

There is also a phase detector which is connected to the charge pump. The phase detector outputs change the VCO frequency in smaller increments and fine tune the VCO.
Figure 4-1: Block diagram of the frequency detector, charge pump, and VCO.

Because the phase detector outputs change the VCO frequency in much smaller increments than the frequency detector outputs, the frequency detector output overwhelms the phase detector output. Thus, the phase detector only affects the VCO when the frequency detector output is quiet. This occurs when the frequency error between the VCO and the data is less than 200kHz. At that point, the original frequency detector can no longer detect a frequency difference between the two signals, and the phase detector locks the VCO to the data.

The frequency detector is said to have “fallen asleep”, which is a desirable feature. The phase detector does all the work now, and the frequency detector does not interfere with it. If the frequency detector outputs are still active, the phase detector cannot lock the VCO to the data and the circuit will not work. This is the same principle that is used to connect the new frequency detector to the charge pump.

4.2 Integrating the Original and New Detectors

Before integrating the two frequency detectors, it is useful to see the relative strengths of the two detectors. Simulations were run with both frequency detectors working in parallel, to see how their outputs respond to the same stimulus. After running the simulations, it is apparent how to best combine the two detectors.

4.2.1 Simulation of the Two Detectors

All the simulations were done with the same conditions as earlier simulations; a 500ns simulation time using a 1.608ns data rate and a sweep of the clock frequency. The number of hits from each detector is plotted vs. the frequency error.

The true output of the VCO-Fast detector is used in these plots, not the output of the
first stage, shown in Figures 3-17 to 3-19. Remember that the output of the VCO-Fast detector goes high after looking at several data edges in a row. For this simulation, I have arbitrarily set the number of flops in the “chain of flops” to be six. The sensitivity of the detector can be increased or decreased by adjusting the number of flip-flops in the chain, and this could be looked at in future work for characterizing this circuit.

MAX Data Pattern

The results of the simulation using a MAX data pattern are shown in Figure 4-2. The plot shows that the new frequency detector has a much wider range than the original detector. The new detector outputs many more hits than the original detector for larger frequency errors, so there should not be any trouble in using these two detectors together. The false hits of the original detector will not interfere significantly with the correct hits from the new detector.

128-bit PRBS Data Pattern

The data for this simulation is plotted in Figure 4-3. The VCO-Fast output dominates the plot. As long as the detection ranges for the original detector and the new detector overlap,
the two frequency detectors will work together to properly lock the VCO. Again, the false hits from the original detector will not adversely affect the new detector.

Because the VCO-Fast output skews the scaling for the plot, Figure 4-4 was generated to give a closer look at the interaction between the VCO-Slow detector and the original detector. As the original detector loses its ability to properly detect the frequency error between the data and the clock, the VCO-Slow detector dominates and provides a clear answer.

**15% Bimodal Jitter With 128-Bit PRBS Data Pattern**

The final simulation incorporates the 15% bimodal jitter on top of the 128-bit PRBS data pattern. The simulation results are shown in Figure 4-5. The VCO-Fast detector suffers greatly from the bimodal jitter, and there is a period of frequency error between 35% and 40% where the original detector will give false hits and the VCO-Fast detector is not active. This is somewhat predicted from Figures 3-18 and 3-19, the simulation data results for the front end of the VCO-Fast detector. These two plots show that the sensitivity of the VCO-Fast detector considerably decreases due to the bimodal jitter, and the detector requires a much larger frequency error to saturate. However, once the VCO-Fast detector begins to
Figure 4-4: A closer look at the 128-bit PRBS data pattern simulation.

Figure 4-5: Simulation of both detectors using 15% bimodal jitter in conjunction with a 128-bit PRBS data pattern.
Figure 4-6: Modified block diagram for PLL incorporating the new frequency detector

activate, it will dominate as it has in the previous two simulations. This condition is very
dangerous, though, and it should be resolved in future work.

From the simulations, it appears that the two detectors can co-exist very well. The new
frequency detector never gives any false hits, and when it is active, it dominates over the
original frequency detector. There is no need to scale the gains of the frequency detectors;
the new one clearly has many more hits than the original for large frequency errors.

4.2.2 Combining the Outputs

To integrate the new detector with the original detector, it is clear from the simulations
that an OR gate between the two outputs is all that is needed. Because the new detector
does not give any false hits, directly adding its output with the original detector will have
the desired effect. The new block diagram for the PLL is shown in Figure 4-6.

This system works on the same principle as the interface between the phase detector and
the original frequency detector. When the new frequency detector is active, it creates many
more output pulses than the original detector, so false hits from the original detector will
not affect the new detector. As the frequency error gets smaller, the new detector starts to
fall asleep, allowing the original detector to take control of the charge pump and guide the
VCO to the correct frequency. As the frequency error becomes even smaller, the original frequency detector falls asleep, allowing the phase detector to fine tune the VCO and finally lock its frequency to the data rate.
Chapter 5

Conclusions

5.1 Problems with the New Frequency Detector

The new frequency detector was designed to be able to detect a wide range of frequency errors. It is not particularly sensitive to medium frequency errors (less than ±20%) so the original detector is still necessary as an acquisition aid.

5.1.1 Performance of the VCO-Slow Detector

The VCO-Slow detector works properly in simulation. It can always be modified to be more or less sensitive by adding or subtracting parallelism from the circuit. The circuit should ideally be able to lock the VCO to the data, but aperture distortion caused by the RESET signal makes it difficult to achieve this. A new method of implementing a RESET in the latches will improve the accuracy of this detector.

The RESET is the important function here, because it only allows the circuit to look for frequency errors within one clock period. As mentioned in Chapter 2, this eliminates the aliasing by keeping track of time.

The essence of the VCO-Slow detector is to look for two data transitions within one clock period. As long as this function is implemented correctly, the detector will work.

5.1.2 Performance of the VCO-Fast Detector

The VCO-Fast detector is difficult to implement correctly, because it depends on the prior state of the system and also relies on a piece of one-bit wide data being transmitted during
some specific time period. There may be more elegant ways of detecting that the VCO is faster than the data rate, but the missing edges in the data stream make it a challenge to design one. The circuit, as described in Section 3.2, is dependent upon the propagation delay of an AND gate. This is an undesirable feature because the propagation delay is not an easily controllable quantity. Future work in this area would be to redesign the front-end of the detector to either eliminate the AND gate, or the dependence upon its propagation delay.

One way to do this is to latch the inputs coming into the AND gate. Once this is done, the AND gate has plenty of time to settle to its correct value, before its output is transferred into the chain of flops. This is a simple thing to do, and removes all the delay associated with the AND gate. Unfortunately, having no propagation delay is not desirable as well, because the circuit in Figure 3-15 relies upon some delay between the front-end and the chain of flops. This was done to compensate for the aperture distortion due to the RESET functions in the flip-flops. Future work in this area could include the design of a stable propagation delay element, or a redesign of the front-end of the detector.

Another area of future work would be to investigate the effects of changing the length of the chain of flops. It is intuitive that using more flip-flops to record the previous state will result in a detector that is less sensitive to smaller frequency errors. Exactly how the sensitivity correlates to the chain length is another matter, though. It would be interesting to investigate this topic, to see if an ideal chain length, or range of lengths, can be determined.

The basic idea in the VCO-Fast detector is to make sure that many bits are more than one clock period wide, and if they are, to signal that the VCO is fast. If a faster process is used, or the system clock is slowed down, a more ideal VCO-Fast detector can be designed and implemented, using any 3 edges of the I or Q clock. This circuit was mentioned in section 3.2.1.

### 5.2 Alternative Methods of Frequency Detection

The wide-range frequency detectors discussed in this thesis are only one solution that can be used for locking the VCO to the data rate. Several alternative methods could be to ramp the VCO input voltage slowly until the existing frequency detector begins to lock onto the signal. The problem with this method is the possibility of harmonic locking, and its slow
acquisition time.

Another method could be to use lockout mechanisms in conjunction with the new detectors to increase sensitivity to certain flavors of frequency error. For example, in Figure 4-5 it was shown that between 30% and 40% frequency error, there was a period where the new detector was not activated and the existing detector was giving false hits. A lockout mechanism could be used such that a pulse from the VCO-Fast detector will disable the FUP output to remove all the false hits from that detector, and allow the original detector to bring the VCO frequency down. Because lockout mechanisms in the past have been used with mixed results, it would be desirable to avoid them if possible.

5.3 Final Conclusions

From the work done on my thesis, I have come to the conclusion that it is possible to build a wide range frequency detector and implement it on an existing process today.
Appendix A

Figures

This appendix contains the schematics for the VCO-Fast and VCO-Slow detector used in the simulations. The schematics were created using Analog Devices’ TC software.
Figure A-1: The VCO-Slow detector.
Figure A-2: The VCO-Fast detector.
Figure A-3: The VCO_SLOWDETV2.1 block. It is the major building block in the VCO-Slow detector.
Figure A-4: The XFAST_LATCHWLVLISHFT block. It is a latch with a level shifted output.
Figure A-5: The OR4INLOW block. It is a simple 4 input OR gate.
Figure A-6: The OR4IN block, a five input OR gate.
Figure A-7: The XFAST_FLOP block. It is the standard flip-flop used in the VCO-Fast detector.
Figure A-8: The XFAST.LATCH block. The basic latch used in the XFAST.FLOP.
Figure A-9: The XFAST_FLOPWAND block. It is a flip-flop combined with an AND gate as its input.
Figure A-10: The XFAST.LATCHWAND block, used in the XFAST.FLOPWAND.
Figure A-11: The XFAST_FLOPWCLR block. It is a flip-flop with a CLEAR function.
Figure A-12: The XFAST.LATCHWCLR block, used in the XFAST.FLOPWCLR.
Figure A-13: The XFMAND block, a 2 input AND gate.
Figure A-14: A multiple input AND gate.
Bibliography


THESIS PROCESSING SLIP

FIXED FIELD: ill. ___________________ name ___________________

index ___________________ biblio ___________________

► COPIES: Archives □ Aero □ Dewey □ Eng □ Hum □
   Lindgren □ Music □ Rotch □ Science □

TITLE VARIES: □

________________________________________

NAME VARIES: □

________________________________________

IMPRINT: (COPYRIGHT) ___________________

________________________________________

► COLLATION: 6SP

________________________________________

► ADD: DEGREE: _______ ► DEPT.: _______

SUPERVISORS:

________________________________________

________________________________________

________________________________________

________________________________________

________________________________________

________________________________________

NOTES:

cat'r: ___________________ date: ___________________ page: J151

► DEPT: E.E. □

► YEAR: 1999 □ ► DEGREE: M.Eng. □

► NAME: PAIK, Steven Sung Hwan □