



Universidad Autónoma de Querétaro

Campus San Juan del Río

Facultad de Ingeniería

Maestría en Ciencias Mecatrónica

“Plataforma de procesamiento paralelo basado en FPGA con sincronización TUC”

TESIS

Que como parte de los requisitos para obtener el grado de Maestra en Ciencias
Mecatrónica

PRESENTA:

Ing. María Amparo Sánchez Magaña

Dirigido por:

Dr. Luis Morales Velázquez

San Juan del Río, Querétaro, abril de 2019



Universidad Autónoma de Querétaro

Campus San Juan del Río

Facultad de Ingeniería

Maestría en Ciencias Mecatrónica

Plataforma de procesamiento paralelo basado en FPGA con sincronización TUC

TESIS

Que como parte de los requisitos para obtener el grado de Maestro en Ciencias Mecatrónica

Presenta:

Ing. María Amparo Sánchez Magaña

Dirigido por:

Dr. Luis Morales Velázquez

SINODALES

Dr. Luis Morales Velázquez
Presidente

Dr. Roque Alfredo Osornio Ríos
Secretario

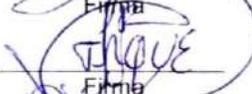
Dr. Miguel Trejo Hernández
Vocal

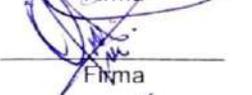
Dr. Jesús Rooney Rivera Guillén
Suplente

Dr. Arturo Yosimar Jaen Cuellar
Suplente

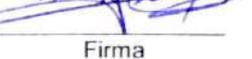
Dr. Manuel Toledano Ayala
Director de la Facultad de Ingeniería

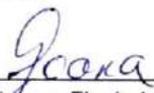

Firma


Firma


Firma


Firma


Firma


Dra. Ma. Guadalupe Flavia Loarca Piña
Directora de Investigación y Posgrado

Querétaro

Abril de 2019

RESUMEN

Actualmente, los sistemas de posicionamiento global (GPS) han tomado gran relevancia en el tema de sincronización en tiempo, ya que estos dispositivos logran generar estampas de tiempo que alcanzan los microsegundos e inclusive nanosegundos de precisión. Estas estampas de tiempo permiten que los eventos en un sistema ocurran siempre al mismo tiempo e incluso se logra conocer con exactitud el momento en que dichos eventos ocurrieron, siendo éstos medidos a partir de un reloj universal. Para lograr la sincronización de un sistema al tiempo universal coordinado es necesario de un elemento que relacione la señal provista del dispositivo GPS y lo transforme como información o como señal de referencia al sistema a sincronizar. Debido a que las sincronizaciones en tiempo requieren no sólo la sincronización en fase, sino también en frecuencia, los lazos de seguimiento de fase son ampliamente utilizados como herramienta para mantener dos señales sincronizadas. En este trabajo se propone el diseño de un lazo de seguimiento de fase completamente digital para la sincronización de un sistema de adquisición de datos mediante la señal provista por un dispositivo GPS. Se estableció una frecuencia para la adquisición de datos, la cual fue mil veces mayor que la frecuencia de la señal provista por el GPS, por lo que el lazo de seguimiento de fase tuvo que diseñarse como sintetizador de frecuencia. Por último, se realizaron pruebas de adquisición sin la implementación del módulo de sincronización para luego comparar estadísticamente dichos resultados contra los obtenidos con la integración del diseño propuesto.

Palabras Clave: Sistemas de posicionamiento global, tiempo universal coordinado, lazo de seguimiento de fase, sincronización en tiempo.

SUMMARY

Currently, Global Positioning Systems (GPS) have taken great relevance in time synchronization because these devices can generate time stamps that reach microseconds and even nanoseconds of precision. These time stamps allow the events in a system to occur at the same time, and it is even possible to know exactly when those events occurred being measured from a universal clock. To achieve the synchronization from a coordinated universal time into a system it is necessary to have an element that relates the signal provided by the GPS device and transform it as information or as a reference signal to the system to be synchronized. Because time synchronizations require not only synchronization in phase, but also in frequency, phase locked loops are widely used to keep two signals synchronized. In this work, we propose the design of an all-digital phase locked for the synchronization of a data acquisition system by means of a signal provided by a GPS device. A frequency was established to the acquisition data, it was thousand times greater than the frequency of the signal provided by the GPS, so the phase locked loop had to be designed as a frequency synthesizer. Finally, acquisition tests were performed without the implementation of the synchronization module to be statistically compared to the results obtained with the integration of the proposed design.

Key Word: Global Positioning Systems, coordinated universal time, phase locked loop, time synchronization.

DEDICATORIA

A todas las personas que hicieron realidad este logro.

AGRADECIMIENTOS

A mi familia, por su amor y apoyo.

A la Universidad Autónoma de Querétaro, por permitirme concluir mis estudios en esta institución.

A mis maestros, por guiarme en mis años como estudiante, especialmente al Dr. Luis Morales por orientarme y ayudarme en todas las ocasiones en las que lo necesité.

A mis amigos por todas las vivencias.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT) por otorgarme la beca número 621053, que me permitió concluir mis estudios de posgrado.

ÍNDICE

1.	Introducción.....	12
1.1	Antecedentes	13
1.2	Descripción del problema.....	16
1.3	Justificación	17
1.4	Hipótesis y objetivos	17
1.4.1	Objetivo general	18
1.4.2	Objetivos particulares	18
1.5	Planteamiento general	18
2.	Fundamentación teórica.....	21
2.1	El problema de la variación de frecuencia	21
2.2	Lazo de seguimiento de fase (PLL).....	24
2.2.1	Comparador digital de fase frecuencia	26
2.2.2	Lazo de filtro	30
2.2.3	Oscilador digital controlado	32
2.3	Dispositivos fpga	32
2.3.1	Los cinco beneficios de la tecnología FPGA	33
2.4	Distribución normal	34
3.	Metodología	36
3.1	Detector de flancos	37
3.2	Contadores y cálculos de factores	38
3.3	Oscilador digital controlado (DCO).....	41
3.4	Divisor de frecuencia.....	42
3.5	Detector de fase frecuencia	43
3.6	Rangos de frecuencias de operación.....	45
4.	Experimentación y resultados	47
4.1	Adquisición no sincronizada.....	48
4.2	Adquisición sincronizada.....	52
4.2.1	Adquisición sincronizada continua.....	52

4.2.2	Adquisición sincronizada con interrupciones	55
4.3	Medición de error de fase	57
4.4	Análisis de resultados	59
5.	Conclusiones.....	63
6.	Bibliografía	65
7.	Anexos	67
7.1	Publicación de artículo	67
7.2	Constancia de presentación de artículo	85
7.3	Máquina de estados, detector de fase frecuencia.....	86
7.4	Módulo detector de fase frecuencia	87

ÍNDICE DE FIGURAS

Figura 1.1 Diagrama a bloques del planteamiento general.	19
Figura 2.1 Variación de frecuencia.....	21
Figura 2.2 Representación gráfica del jitter.	23
Figura 2.3 Esquema general del lazo de seguimiento de fase.	25
Figura 2.4 Diagrama a bloques del PFD.	26
Figura 2.5 Comportamiento de la salida según las entradas en el PFD.....	28
Figura 2.6 Formas de onda a la salida del PFD. a) Cuando el error de fase es cero. b) Cuando el error de fase es positivo. c) Cuando el error de fase es negativo. ...	29
Figura 2.7 Promedio salida del PFD contra error de fase.	30
Figura 2.8 Diagrama del lazo de filtro digital.	30
Figura 2.9 Formas de onda de la salida del filtro de lazo.	31
Figura 2.10 Diagrama del oscilador digital controlado.....	32
Figura 2.11 Función de distribución Gaussiana.	35
Figura 3.1 Diagrama de ADPLL propuesto.	36
Figura 3.2 Diagrama interno del detector de flancos.....	37
Figura 3.3 Salida del contador ascendente contra el error de fase.	39
Figura 3.4 Diagrama de flujo del detector fase frecuencia.	44
Figura 3.5 Rangos de operación en el PLL.	46
Figura 4.1 Esquema general de la puesta del experimento.	47
Figura 4.2 Señales adquiridas sin sincronización al inicio de la adquisición.	48

Figura 4.3 Señales adquiridas sin sincronización después de 30 s de adquisición continua.....	49
Figura 4.4 Señales adquiridas sin sincronización luego de 250 s de adquisición continua.....	51
Figura 4.5 Señales adquiridas sin sincronización luego de 496 s de adquisición continua.....	51
Figura 4.6 Señales envolventes del error sin sincronización.....	52
Figura 4.7 Señales adquiridas con sincronización al inicio de la adquisición.....	53
Figura 4.8 Señales adquiridas con sincronización luego de 250 s de adquisición continua.....	53
Figura 4.9 Señales adquiridas con sincronización luego 496 s de adquisición continua.....	54
Figura 4.10 Señales envolventes del error con sincronización.....	54
Figura 4.11 Envolventes del error con interrupción de 1 segundo.	55
Figura 4.12 Envolventes del error con interrupción de 30 segundos.....	56
Figura 4.13 Envolventes del error con interrupción de 60 segundos.....	56
Figura 4.14 Medición de jitter con osciloscopio para señal sin sincronización.	58
Figura 4.15 Medición de jitter con osciloscopio para señal con sincronización.	58
Figura 4.16 Funciones de distribución normal obtenido de envolventes del error en las adquisiciones con y sin sincronización.	62

ÍNDICE DE TABLAS

Tabla 2-1 Combinaciones de salidas en el PFD.....	27
Tabla 3-1 Valores de los factores calculados para el diseño del ADPLL.	41
Tabla 3-2 Tabla de verdad del DCO.....	42
Tabla 4-1 Parámetros estadísticos calculados.	60

1.INTRODUCCIÓN

Es bien conocido que los equipos electrónicos como los microprocesadores, o microcontroladores utilicen elementos oscilantes para su funcionamiento; a este elemento se le conoce como oscilador de frecuencia e indica la velocidad a la que el sistema debe operar. Los osciladores son capaces de generar una señal periódica sin la necesidad de ninguna señal de referencia, es decir, la señal periódica se genera simplemente con una señal de entrada de corriente directa (Tomasi, 2003). Todos los osciladores trabajan a partir de una frecuencia nominal que no es más que la frecuencia ideal que se desea a la salida del oscilador. Sin embargo, se ha demostrado que el comportamiento de estos dispositivos electrónicos se encuentra estrechamente ligado a sus defectos de fabricación, cambios en la temperatura del entorno, interferencias magnéticas y eléctricas que hacen que la frecuencia real de operación varíe con respecto a la especificada por el proveedor (Pérez Vega *et al.*, 2007). Estas variaciones de frecuencia se presentan en la mayoría de los osciladores y pueden llegar a ser muy pequeñas; sin embargo, incluso la variación más pequeña en la frecuencia del oscilador agrega errores de fase en la señal generada que se van acumulando y aumenta respecto al tiempo.

Para ejemplificar mejor los efectos de la variación de frecuencia en largos periodos de trabajo en los osciladores se consideran dos relojes que trabajan a partir de un oscilador de cuarzo, independiente uno del otro. Al inicio, ambos relojes se encuentran sincronizados, pero por cada segundo transcurrido, existe una diferencia entre ellos de 0.1 milisegundos. Pese a que pareciera una cantidad casi despreciable, luego de un año de operación continua ambos relojes difieren en tiempo más de cincuenta minutos. Es por esta razón que la sincronización en tiempo toma cada vez más relevancia en distintas áreas, sobre todo en materia de telecomunicación, ya que gracias a la sincronización se puede contar con una referencia de tiempo única.

En la actualidad existen distintos sistemas de tiempo que son globalmente utilizados para proporcionar una referencia de tiempo precisa. Entre estos se encuentra el Tiempo Universal Coordinado (UTC - *Coordinated Universal Time*), el cual es el más utilizado dentro de los dispositivos GPS y el Tiempo Atómico Internacional (TAI - *International Atomic Time*), cuyas escalas de tiempo se encuentran desfasados por exactamente 32 segundos. En el año de 1980, la escala de tiempo en los dispositivos GPS fueron establecidos utilizando el Tiempo Universal Coordinado, lo cual permite que estas escalas no presenten saltos de tiempo. Es debido a esto que los dispositivos GPS son cada vez más utilizados como referencia en escalas de tiempo o para proveer de estampas de tiempo exactas (El-Rabbany, 2002).

A pesar de que el GPS provee de estampas de tiempo exactas, es necesario de un módulo de sincronización que logre enlazar las señales provistas con el GPS y las transmita como información al sistema a sincronizar. En la presente tesis se describe el desarrollo del módulo de sincronización para lograr que la señal generada por una tarjeta de arreglos de compuertas programables en campo (FPGA – *Field Programmable Gate Array*) que se encuentre sincronizada a la señal de referencia que provee un dispositivo GPS. En este proyecto se propone un lazo de seguimiento de fase, ya que estos circuitos pueden ser utilizados como sintetizadores de frecuencia para obtener una señal de salida con frecuencia mayor a la de referencia. Se pretende que a partir de la referencia se obtenga una señal con periodo de 1 ms que se mantenga sincronizada con respecto al Tiempo Universal Coordinado.

1.1 Antecedentes

Las ideas básicas de los lazos de seguimiento de fase (PLL's) comenzaron cerca de 1919 y desde entonces éstos han ayudado al desarrollo de sistemas de comunicación, procesamiento y sincronización de señales. La introducción de este nuevo elemento comenzó ante la necesidad de crear un sistema que sincronizara osciladores, y la teoría en la que se basaron los PLL fue a partir de la

retroalimentación, que se comenzaba a aplicar en los amplificadores. No fue hasta el año de 1965, con la introducción de los circuitos integrados (IC's), que los PLLs comenzaron a ser cada vez más utilizados. A partir de este momento, comenzó una era de desarrollo y los lazos de seguimiento lineales comenzaron a trabajar con señales digitales, dando paso a los lazos de seguimiento digitales (DPLL) o también llamados PLL de señales mixtas (Ammar *et al.*, 2014).

La época digital ha permitido que se desarrollen lazos de seguimiento completamente digitales (ADPLL – *All Digital Phase Locked Loop*) y aunque siguen existiendo lazos de seguimiento analógicos, las principales ventajas que proveen los sistemas digitales, sobre el analógico son el bajo consumo de energía, mayor escalabilidad y la reducción del tamaño en el circuito (Ibrahim Afifiy *et al.*, 2014). En los ADPLL se tienen, a su vez, distintas arquitecturas; su comportamiento depende de los bloques que se utilizan en su diseño y su funcionamiento se adapta según la necesidad del diseñador.

Las arquitecturas más básicas utilizan detectores de fase basados en la compuerta lógica tipo XOR. Lata & Kumar (2013) desarrollaron un ADPLL de este tipo, sin embargo, la mayor limitación del sistema era el corto rango de detección en el error entre las señales. El rango está directamente relacionado al detector de fase que se implementa y éste opera entre los -90 y 90 grados. Un segundo inconveniente se presenta al manejar un bloque asíncrono, ya que debido a esto y a la implementación en una tarjeta FPGA se introduce un retardo gracias a la compuerta utilizada (Edway & Manjunath, 2017).

Ante la necesidad de aumentar el rango de detección en los ADPLL, se desarrollaron los detectores de fase utilizando *flip-flops* tipo JK. Gayathri (2013) desarrolló un ADPLL en lenguaje HDL construido sólo de circuitos lógicos. Al eliminar la parte analógica del sistema, se eliminaron errores de variación en los componentes, derivadas del envejecimiento de los mismos. En este trabajo se detalla un detector de fase tipo JK. A diferencia de los trabajos hasta ahora mencionados, a este detector de fase se le agregó una tercera señal de entrada, la

señal del reloj principal; el oscilador digital controlado también trabaja a partir de la señal del reloj principal, por lo que se obtuvo un sistema completamente síncrono. El cambio del detector de fase resultó en un aumento en el rango para la detección de error. Por último, se logró sincronizar la señal generada no sólo en fase, sino también en frecuencia.

Vydeki & Srinivasa (2014) utilizaron un detector de fase utilizando el *flip-flop* tipo JK, pero además agregaron un módulo contador al diseño, esto añadió sensibilidad a los flancos ascendentes de las señales, es decir, con esto lograron que el sistema fuera síncrono. No obstante, el contador se reinicia en cada flanco ascendente de la señal de referencia, por lo que la salida del detector de fase permanece en cero siempre que la señal de referencia esté en atraso a la señal generada por el ADPLL.

En los últimos años, los sistemas de posicionamiento global han sido recurrentemente utilizados para mejorar la sincronización de señales entre nodos con el fin de evitar deslices que pueden ser perjudiciales en los servicios de datos. Gasparini *et al.* (2007) desarrollaron un circuito digital implementado en FPGA con el fin de reducir el error de fase presente en una señal generada por dicha tarjeta. La señal de entrada utilizada como referencia para el sistema era un pulso obtenido de un dispositivo GPS. La alta precisión de este dispositivo permitió que el error de fase disminuyera en más del 60%. El mayor reto del proyecto consistió en obtener una señal de mayor frecuencia a la salida del sistema de sincronización, para lograrlo se tuvo que desarrollar un controlador PI (Proporcional-Integral) que basaba su arquitectura en un lazo de seguimiento de fase digital, lo cual resultó en tiempo de bloqueo de la señal muy corto.

Así mismo, Monteiro *et al.* (2016) lograron la sincronización en tiempo global mediante la implementación de un controlador PI retroalimentado con el error de fase de un pulso por segundo. Al diseño original del controlador se agregó módulos que permitieron la sincronización aún cuando la señal del GPS no existiese. Cuando esto sucedía, el cálculo del error de fase se obtuvo mediante una señal virtual

de referencia generada por la tarjeta FPGA. La señal obtenida del sistema de sincronización logró errores de fase menores a 100 ns, sin embargo, el controlador no logró un tiempo de respuesta menor a dos minutos.

1.2 Descripción del problema

Actualmente se cuenta con una red de dispositivos que adquieren una señal en distintas ubicaciones; estos dispositivos se mantienen trabajando durante largos periodos de operación continua y se ha detectado una discrepancia entre las mediciones proporcionadas de una tarjeta y otra. Estas diferencias son debidas a las variaciones de frecuencia presente en el oscilador de las tarjetas y una solución a este problema es utilizar osciladores atómicos que trabajen como referencia para generar una señal para la adquisición sincronizada, sin embargo, se debe tener en cuenta que dichos osciladores de frecuencia también presentan una deriva luego de un largo tiempo, por lo que al fin y al cabo se requeriría de un reajuste, además de que el uso de estos relojes no garantizarían la sincronización con otros relojes, es decir, se deben alinear dichas referencias en un inicio para que todos los dispositivos se encuentren en sincronía.

Dentro de los trabajos referenciados se explicó que el dispositivo comúnmente utilizado para la sincronía de señales es el GPS, sin embargo, este dispositivo es propenso a perder la conexión con la red satelital debido a interferencias del medio no controladas. Otro reto a considerar es entonces, el lograr la sincronización de varios dispositivos con osciladores independientes que mantenga la sincronización aún y cuando la señal de referencia del GPS se haya perdido durante un cierto periodo de tiempo.

1.3 Justificación

Los sistemas de posicionamiento global mantienen una conexión con la red satelital y logran sincronizarse con tres o cuatro relojes atómicos monitoreados y controlados para mantenerse sincronizados durante todo momento. Es por esta razón que son una gran herramienta en sistemas de sincronización, ya que logran alcanzar la precisión de un reloj atómico eliminando la necesidad de un reloj atómico local. De esta manera, al utilizar un dispositivo GPS como elemento para la sincronización se evitaría un reajuste manual de los osciladores en las tarjetas. Aunado a esto, los receptores GPS no sólo proveen de una referencia de tiempo exacta, sino que además es absoluta, es decir, está alineado con el tiempo universal coordinado.

Por otra parte, es necesario que en el proceso de sincronización se logre aumentar la frecuencia de resolución de la señal de referencia. Si bien el pulso generado por el GPS es exacto, éste alcanza frecuencias de sólo 1 Hz. Los lazos de seguimiento de fase no sólo sincronizan dos señales de la misma frecuencia, sino que además podemos obtener a la salida una señal de frecuencia mayor que posee poca fluctuación. Aunque existen varios tipos de lazos de seguimiento de fase que encuentran ya documentados, es necesario realizar cambios en el diseño del mismo con el fin de mantener la sincronización aún y cuando el GPS haya perdido el enlace con la red satelital.

1.4 Hipótesis y objetivos

Mediante la implementación de un lazo de seguimiento de fase y una referencia de tiempo con baja fluctuación proporcionada por un dispositivo GPS, se obtendrá una señal de frecuencia mil veces mayor que posea las mismas características de estabilidad que el oscilador principal.

1.4.1 Objetivo general

Desarrollar un sistema de procesamiento paralelo de señales en tiempo real basado en FPGA considerando la sincronización de datos con el tiempo universal coordinado.

1.4.2 Objetivos particulares

- Implementar y habilitar la antena de recepción GPS en la plataforma FPGA para obtener la señal de referencia en base al Tiempo Universal Coordinado.
- Diseñar e implementar un lazo de seguimiento de fase completamente digital con el fin de obtener una señal de 1 kHz que mantenga las mismas condiciones de estabilidad que el oscilador de la tarjeta DUA 4.
- Seleccionar un convertidor analógico digital para un sistema de adquisición de datos para la validación del módulo de sincronización.
- Acondicionar los datos obtenidos para posteriormente ser procesados y de esta manera corroborar el funcionamiento del diseño propuesto.
- Medir y comparar el error de fase de la señal de 1 kHz obtenida con y sin sincronización.
- Obtención de parámetros estadísticos para la validación del sistema de sincronización.

1.5 Planteamiento general

El planteamiento general de la metodología a realizarse se muestra en la Figura 1.1, la cual consiste en cinco módulos principales que son descritos a continuación:

1.- Habilitación antena GPS: El receptor GPS se configura a través de mensajes de consulta-respuesta mediante comunicación basada en mensajes NMEA (National Marine Association), los cuales utilizan el protocolo UART (Universal Asynchronous Receiver-Transmitter – Transmisor-Receptor Asíncrono Universal) como medio de comunicación. Con esto se puede acceder a la configuración para obtener un pulso cada segundo (PPS).

2.- ADPLL: Es el sistema de sincronización propuesto en el presente trabajo, el cual genera una señal de mayor resolución a la obtenida por la antena GPS y que elimina el error de fase acumulativo en dicha señal.

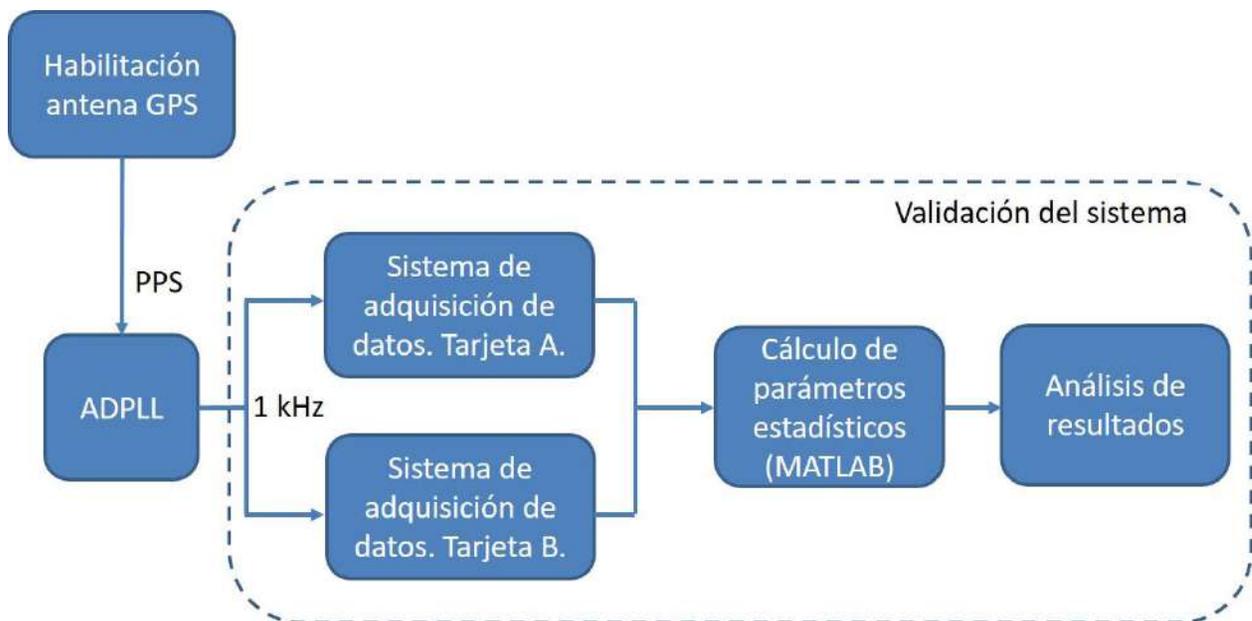


Figura 1.1 Diagrama a bloques del planteamiento general.

3.- Sistema de adquisición de datos: Este bloque está compuesto por un circuito para la conversión de señales analógicas a su equivalente digital. Se encuentra implementado en dos tarjetas DUA 4.

4.- Cálculo de parámetros estadísticos: Los resultados obtenidos en la adquisición se emplean para calcular el error entre las señales adquiridas, promedio del error, desviación estándar de error, valor RMS del error, entre otros.

5.- Análisis de resultados: Gracias a la obtención de los valores en el bloque anterior se puede realizar un análisis para determinar la eficacia del diseño propuesto.

2.FUNDAMENTACIÓN TEÓRICA

2.1 El problema de la variación de frecuencia

Para mostrar el efecto de las variaciones de frecuencia en los osciladores, se toman como ejemplo las señales de la Figura 2.1, en ella se observan tres señales de onda cuadradas generadas por un oscilador. En la primera señal se muestra un comportamiento ideal; es decir, cada uno de los diez ciclos generados posee la misma frecuencia, a la cual se le denomina frecuencia nominal. Para f_1 , la variación de frecuencia ocurre cada tres ciclos y los ciclos 3, 6 y 9 poseen, aunque casi indetectable, un decremento en su frecuencia con respecto a la frecuencia nominal. Por último, la señal f_2 cuenta con un incremento en la frecuencia en los ciclos 3, 6 y 9. Una vez completados los diez ciclos de trabajo, se observa que tanto para la señal f_1 y f_2 el periodo de estas señales no es el mismo que posee la señal f_{nom} , aun cuando las variaciones de frecuencia se presentaron únicamente en 3 ciclos de trabajo.

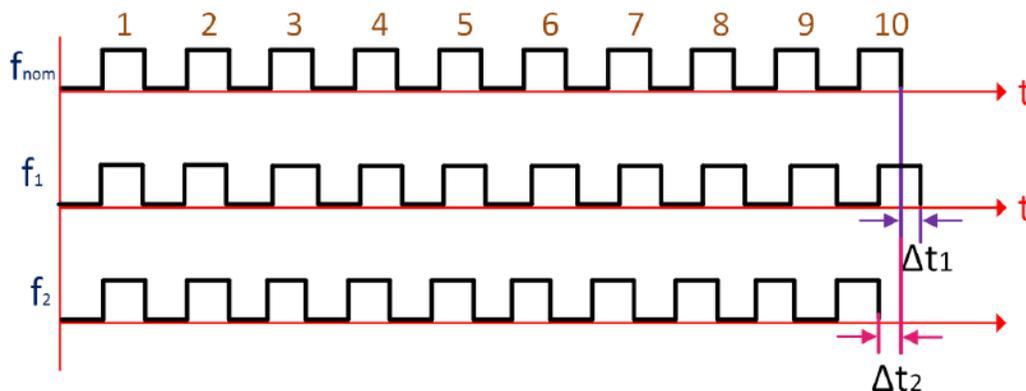


Figura 2.1 Variación de frecuencia.

El fenómeno antes descrito se presenta comúnmente en la frecuencia nominal de los osciladores, la cual se encuentra especificada en la hoja de datos del dispositivo. Dentro las características de operación se indica también la estabilidad del mismo, la cual determina las variaciones de frecuencia máximas que

pueden presentarse entre la frecuencia nominal y la frecuencia real del dispositivo. Para determinar estas variaciones se utiliza el termino partes por millón (Parts-Per-Million, PPM) (Tomasi, 2003).

En el presente proyecto se utiliza una tarjeta en desarrollo DUA 4, la cual utiliza el oscilador 535-9995-1ND de Abracon LLC con una frecuencia nominal de 48 MHz y una variación de frecuencia especificada de +/-100 PPM (Abracon, 2014). Las frecuencias máximas y mínimas de operación se pueden calcular mediante las ecuaciones 1 y 2 respectivamente.

$$f_{max} = \frac{f_{nominal}}{1 + \frac{PPM}{1 \times 10^6}} \quad (1)$$

$$f_{min} = f_{nominal} \left(1 + \frac{PPM}{1 \times 10^6} \right) \quad (2)$$

Para el oscilador con las especificaciones antes mencionadas se calcula una frecuencia máxima de operación $f_{max} = 47,995,200.48$ y una frecuencia mínima de $f_{min} = 48,004,800$ Hz. Las variaciones de frecuencia no sólo provocan en un cambio de la frecuencia nominal del oscilador, sino que además se induce un ruido de fase en la señal generada, también conocida como jitter.

El jitter es una desviación indeseada del periodo de una señal periódica. Esta desviación puede ser de manera positiva, aumentando el periodo o de manera negativa, en la cual se disminuye el periodo nominal de la señal y puede ocurrir tanto en el flanco ascendente como en el descendente. En la Figura 2.2 se muestra gráficamente el comportamiento de una señal generada que posee ruido de fase (McNeill & Ricketts, 2009).

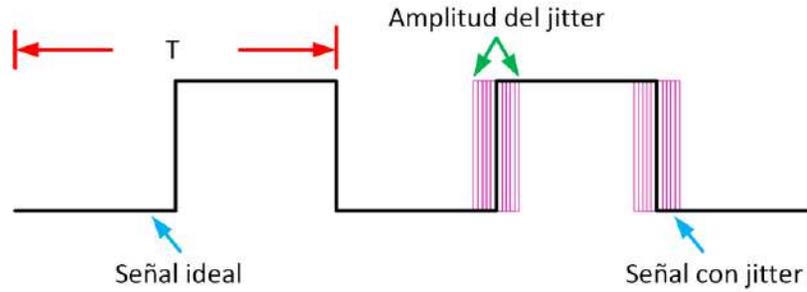


Figura 2.2 Representación gráfica del jitter.

La amplitud del jitter está ligado directamente con la tolerancia del oscilador, y la desviación del periodo varía de manera $T=T\pm T'$, donde T' es un valor que se obtiene mediante las frecuencias mínimas y máximas de operación del oscilador, tal y como lo describen las ecuaciones 3 y 4.

$$\Delta T'_1 = \left| \frac{1}{f_{nom}} - \frac{1}{f_{max}} \right| \quad (3)$$

$$\Delta T'_2 = \left| \frac{1}{f_{nom}} - \frac{1}{f_{min}} \right| \quad (4)$$

donde

$\Delta T'_1$ = variación del periodo a frecuencias máximas de operación.

$\Delta T'_2$ = variación del periodo a frecuencias mínimas de operación.

f_{nom} = frecuencia nominal de oscilación.

f_{min} = frecuencia mínima de oscilación.

$f_{máx}$ = frecuencia máxima de oscilación.

Esta diferencia de periodo en el oscilador de la tarjeta corresponde a $\Delta T'_1 = 2.0831 \times 10^{-12} \text{ s}$ y $\Delta T'_2 = 2.0833 \times 10^{-12} \text{ s}$. Para ambos casos la diferencia de periodos entre la señal ideal y la señal real no sobrepasa ni siquiera el 0.01% de error. Sin embargo, si la señal de oscilación se mantuviera generando sólo a frecuencias mínimas, luego de treinta días de trabajo continuo se obtendría una diferencia de periodos de $\Delta T_2 = 259.14 \text{ s}$ o lo que es igual, la señal real del oscilador se encontraría 4.32 minutos en atraso con respecto a una señal ideal con la misma frecuencia. Por lo que, aunque el error debido a la variación de frecuencia sea pequeño, este es acumulativo y aumenta con el paso del tiempo.

2.2 Lazo de seguimiento de fase (PLL)

Los lazos de seguimiento de fase son circuitos retroalimentados capaces de generar una señal de salida sincronizada tanto en fase como en frecuencia con respecto a una señal de entrada o referencia. En el diagrama a bloques del lazo de seguimiento de fase básico se cuenta con tres bloques principales: el detector de fase-frecuencia, un filtro pasa bajas y un oscilador controlado por tensión (VCO). Dentro de sus aplicaciones se encuentran aquellas en sistemas de comunicaciones, como generadores y sintetizadores de frecuencia, en procesos de modulación y demodulación, filtros de seguimiento, entre otras. Cuando un PLL trabaja como sintetizador de frecuencia, se agrega un bloque adicional al sistema, el divisor de frecuencia, el cual se encarga de dividir la frecuencia de la salida del VCO entre un factor N (Best, 2003). En la Figura 2.3 se muestran los bloques en un PLL que actúa como sintetizador indirecto.

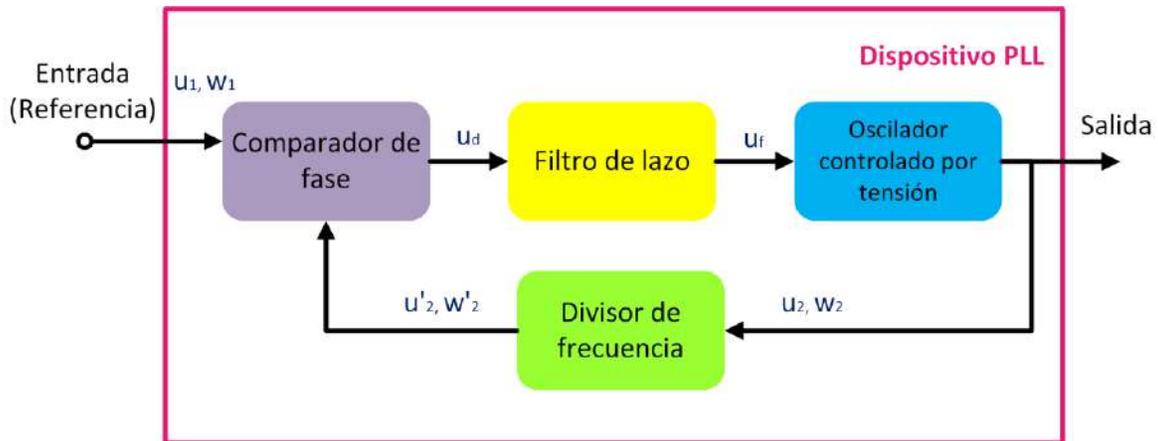


Figura 2.3 Esquema general del lazo de seguimiento de fase.

El proceso de sincronización inicia a partir de una señal de entrada que sirve como referencia al sistema, si dicha referencia no existe, entonces la señal a la salida del circuito será nula. Al aplicar una señal de entrada (u_1), el PLL compara la fase y la frecuencia (w_1) de la señal de referencia con la señal obtenida a la salida del divisor de frecuencia (u'_2). Esta señal posee una frecuencia (w'_2) múltiplo a la frecuencia producida por el VCO (w_2), obteniendo de esta comparación una tensión que es proporcional al ángulo de fase y a la frecuencia entre dichas señales (u_d). Debido a las componentes de alta frecuencia que posee la señal a la salida de la comparación, se puede realizar un filtrado de la señal de error (u_f). El PLL actúa entonces como un mecanismo de control que logra modificar el periodo del VCO de manera que el error de fase sea reducido al mínimo. Si, por otro lado, ambas señales se encuentran alineadas se dice entonces que el lazo de seguimiento se encuentra en un “estado de bloqueo” y el oscilador trabaja a su frecuencia libre de oscilación (también llamada frecuencia central) (Gallardo, 2015).

Los PLL se pueden clasificar según su diseño:

- PLL analógico (APLL): El detector de fase es un multiplicador analógico y las variaciones de frecuencia de la señal generada se logra mediante un oscilador controlado por voltaje (VCO).

- PLL digital (DPLL): Utiliza un detector de fase digital constituido por compuertas XOR, disparadores de flancos JK o detectores digitales de fase frecuencia. Puede contener un divisor digital en el lazo.
- PLL completamente digital (ADPLL): El detector de fase, el filtro pasa bajas y el oscilador son digitales. Utiliza un oscilador digital controlado (DCO) (Best, 2003).

En la sección a continuación se describen los componentes principales en un lazo de seguimiento completamente digital.

2.2.1 Comparador digital de fase frecuencia

Un detector de fase es un circuito capaz de comparar la fase de la señal de referencia con la señal generada por el oscilador controlado. Este módulo proporciona una señal que es directamente proporcional a la diferencia de fases entre las dos señales antes mencionadas.

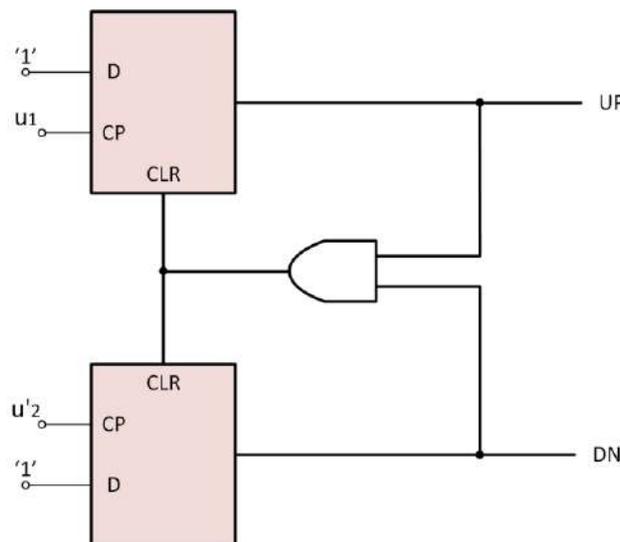


Figura 2.4 Diagrama a bloques del PFD.

En la actualidad existen distintos tipos de detectores de fase y cada uno posee características distintas, siendo el rango de bloqueo la más importante. El detector de fase digital más utilizado es el detector de fase frecuencia (PFD – *Phase Frequency Detector*) ya que además de proporcionar información sobre el error de fase, también es sensible a saltos de frecuencia en las señales. EL PFD está constituido por dos *flip-flops* tipo D y una compuerta lógica *AND*, como se muestra en la Figura 2.4 (Gayathri, 2013).

Debido a que se trata de un PLL trabajando como sintetizador de frecuencia, las señales de entrada en el PFD son la señal de referencia (u_1) y la señal de salida del divisor de frecuencia (u'_2). A la salida se obtienen dos señales, *UP* y *DN*. En la Tabla 2-1 se muestran los cuatro estados que puede tomar el PFD. El cuarto estado se inhabilita, ya que las señales de salida no pueden encontrarse en alto al mismo tiempo y ambos registros se reinician en ese estado.

Tabla 2-1 Combinaciones de salidas en el PFD.

Estado 1	UP=0, DN=0	Estado 0
Estado 2	UP=1, DN=0	Estado -1
Estado 3	UP=0, DN=1	Estado 1
Estado 4	UP=1, DN=1	Estado inhabilitado

Al inhabilitarse el cuarto estado se tiene que la señal de salida del PFD es una señal de tres estados. Para poder obtener esta señal única de salida en el PFD se debe realizar un arreglo de transistores MOS. A este arreglo entran las señales *UP* y *DN* y se obtiene una señal de salida tri-estados, u_d . El estado actual de la salida U_d se puede ver en el diagrama de la Figura 2.5. Se tienen tres estados: cuando ambas señales se encuentran sincronizadas tanto en fase, como en

frecuencia, la señal de salida se encuentra en estado '0', es decir, u_d entra en alta impedancia. Cuando la salida del divisor de frecuencia, u'_2 está en adelanto en referencia a u_1 , la señal de salida toma el estado de -1. Por último, cuando la señal de referencia se encuentra en adelanto a respecto a la señal de retroalimentación entonces la señal de salida toma el estado de +1. En la Figura 2.6 se detalla el comportamiento de la señal de salida del PFD en un diagrama de tiempo (Best, 2003).

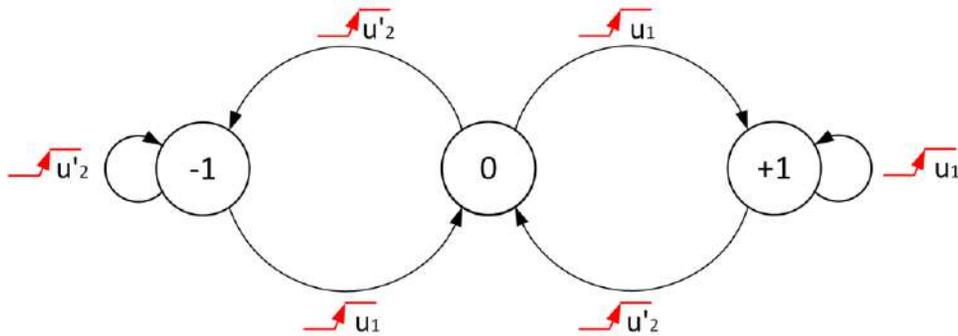


Figura 2.5 Comportamiento de la salida según las entradas en el PFD.

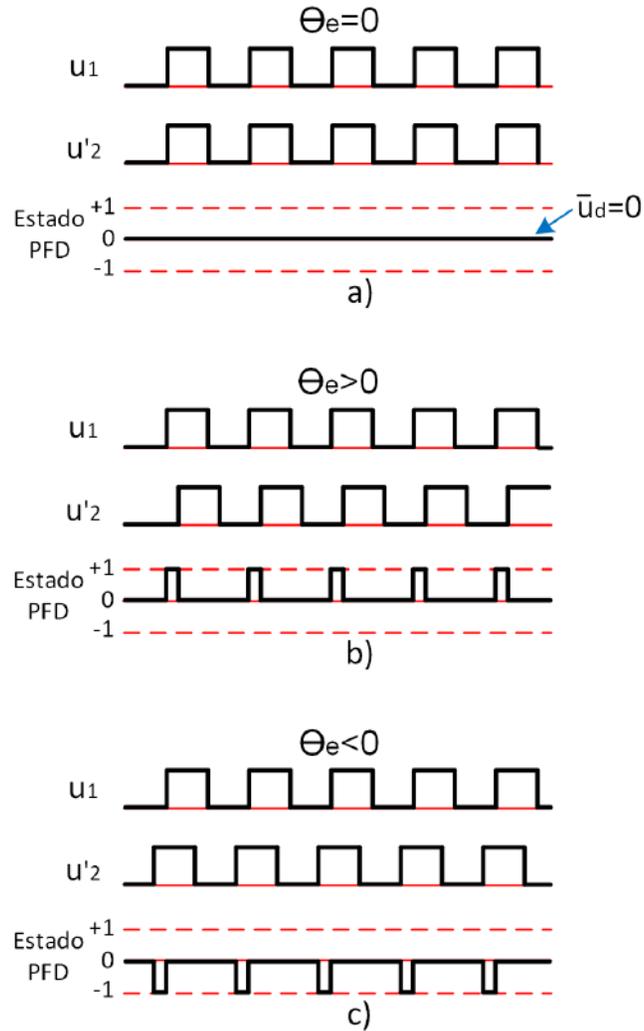


Figura 2.6 Formas de onda a la salida del PFD. a) Cuando el error de fase es cero. b) Cuando el error de fase es positivo. c) Cuando el error de fase es negativo.

Otra ventaja que proporciona el PFD sobre los otros detectores de fase es que posee un rango de detección mucho mayor. El promedio de la señal de salida del PFD (\bar{u}_d) es proporcional en el rango de -2π y 2π tal y como lo muestra la Figura 2.7.

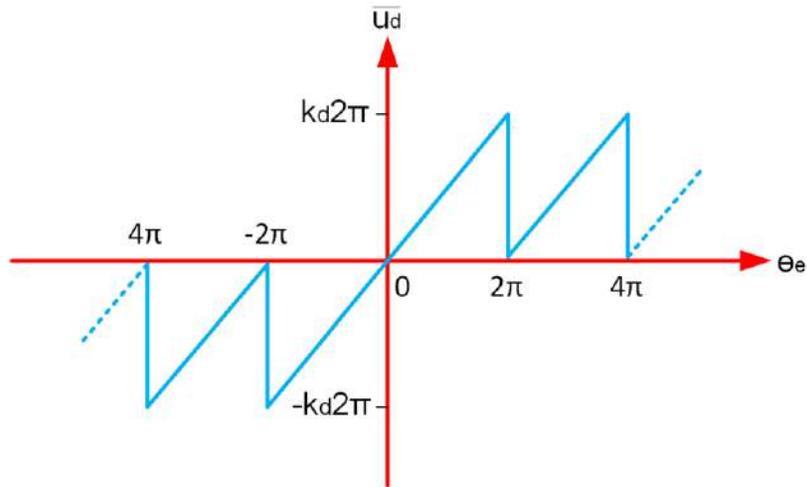


Figura 2.7 Promedio salida del PFD contra error de fase.

2.2.2 Lazo de filtro

Es el análogo de un filtro pasa bajas. No se encuentra presente en todos los ADPLL, pero en lazos de orden mayor, como aplicaciones de servo control o telecomunicaciones los lazos requieren de un filtro si es necesario. El filtro de lazo digital más utilizado es el filtro contador ascendente-descendente. Como su nombre lo indica, se trata de dos contadores que operan en conjunto con el detector de fase frecuencia como se muestra en la Figura 2.8 (Best, 2003).

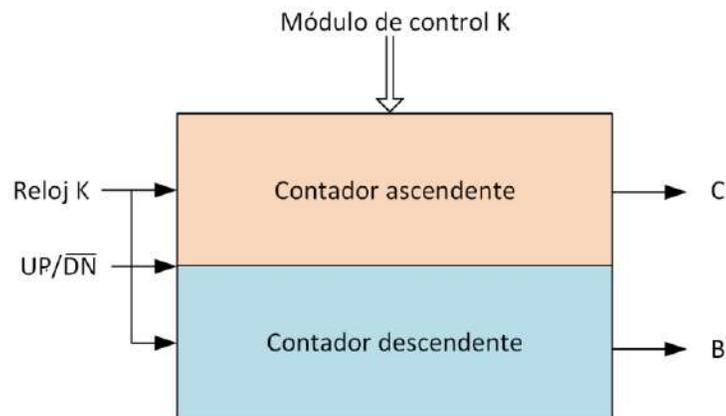


Figura 2.8 Diagrama del lazo de filtro digital.

A pesar de sus nombres, internamente son dos contadores ascendentes. La señal de control K es un valor numérico que entra a ambos contadores por lo que la cuenta en ambos módulos va del rango desde 0 hasta $K-1$. El valor de la señal K debe ser un entero potencia de 2. La señal de *Reloj* K debe ser M mayor que la frecuencia central del ADPLL, K es típicamente 8, 16, 32, y así sucesivamente. La señal *UP/DN* controla el comportamiento de los contadores. Cuando esta señal está en alto, el contador ascendente comienza a contar mientras que el contador descendente mantiene la cuenta. Cuando la señal está en bajo, el contador descendente comienza a contar, mientras que el contador ascendente mantiene la cuenta. Las salidas C y B es el bit más significativo del contador ascendente y el contador descendente, respectivamente (Gayathri, 2013). El diagrama de la Figura 2.9 muestra el comportamiento de las salidas en el filtro de lazo.

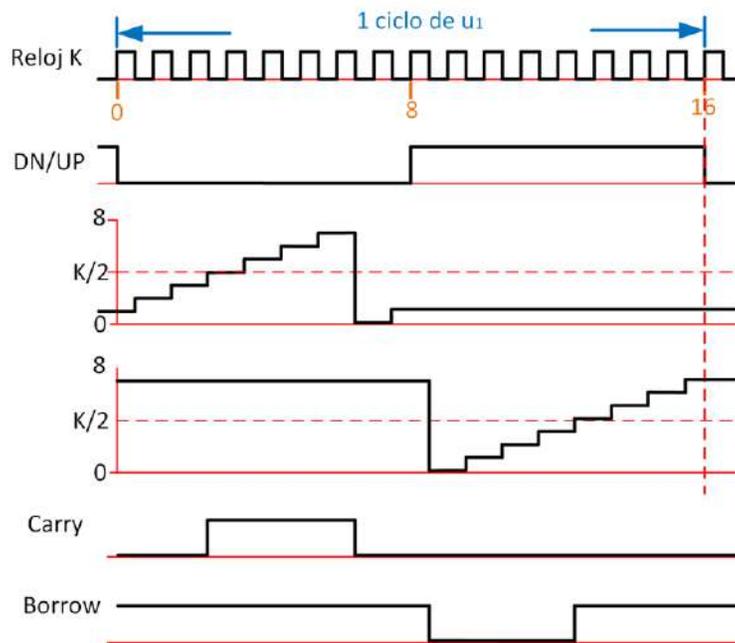


Figura 2.9 Formas de onda de la salida del filtro de lazo.

2.2.3 Oscilador digital controlado

Consiste en un *flip-flop* tipo T o de conmutación (Figura 2.4). Tiene como entradas las señales *C* y *B* que provienen directamente del lazo de filtro y contiene una tercera señal de entrada, el cual es un reloj de entrada; a la salida se tiene un reloj cuyo período es modificado directamente por las señales de entrada *C* y *B*.

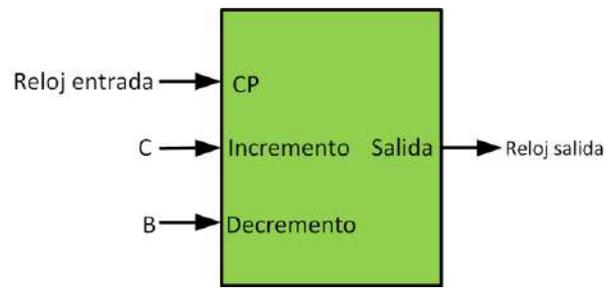


Figura 2.10 Diagrama del oscilador digital controlado.

Los *flip-flops* de conmutación alternan en cada flanco positivo del reloj de entrada y el reloj de salida varía mediante la siguiente función lógica. $\text{Reloj de salida} = \text{NOT}(\text{reloj de entrada}) \text{ AND } (\text{NOT}(\text{salida flip-flop T}))$. La secuencia del reloj de salida cuando *C* y *B* se mantienen en bajo, la secuencia del reloj de salida es “bajo-bajo-alto-bajo-bajo”. Cuando *C* se encuentra en estado lógico alto, la secuencia es de la forma “bajo-bajo-alto-bajo-alto”, es decir, se agrega un ciclo para reducir el período de la señal de salida. En el caso de que *B* se encuentre en estado lógico alto, la secuencia del reloj a la salida es del tipo “bajo-alto-bajo-bajo-bajo”, en este último caso, la secuencia hace que el período a la salida aumente.

2.3 Dispositivos FPGA

Desarrollados por primera vez por la empresa Xilinx en 1985, los arreglos de compuertas programables en campo (FPGA) son chips de silicio reprogramables. La adopción del chip FPGA en las industrias se debe al hecho de que éstos combinan las características de los ASICs (Application Specific Integrated Circuit,

circuitos integrados de aplicación específica) y de los sistemas basados en procesador, es decir, se logra implementar la aplicación personalizada en circuitos de hardware en lugar de ejecutarse sin la necesidad de un sistema operativo, controladores o software de aplicación.

2.3.1 Los cinco beneficios de la tecnología FPGA

Rendimiento: Los FPGAs exceden la potencia computacional de los procesadores digitales de señales (DSP, Digital Signal Processor) rasgando el paradigma de ejecución secuencial y por ende logrando más en cada ciclo de reloj. Al controlar entradas y salidas a nivel hardware se ofrece un mejor tiempo de respuesta y funcionalidad especializada que concuerda con los requerimientos de distintas aplicaciones.

Tiempo en llegar al mercado: La tecnología FPGA brinda la flexibilidad y capacidad necesaria para un rápido desarrollo de prototipos y enfrentar los retos de que un producto se libere tarde al mercado. Una nueva idea o concepto pueden verificarse en hardware sin la necesidad de pasar por el largo proceso de fabricación como sucede para un diseño personalizado en ASIC.

Precio: Los costos de la ingeniería de un diseño personalizado ASIC superan considerablemente a las soluciones de hardware basadas en FPGA. La naturaleza programable del silicio elimina el precio de fabricación y los largos plazos de ejecución de ensamblado. Para un sistema no siempre se tienen los mismos requerimientos, estos cambian con el tiempo y el precio de cambiar los diseños FPGA es insignificante al compararlo con el precio de implementar cambios en un ASIC.

Fiabilidad: Los equipos de software ofrecen un entorno de programación, mientras que los circuitos de un FPGA son una implementación segura de la ejecución de un programa. En aquellas herramientas de software el driver se encarga de administrar los recursos de hardware y el sistema operativo administra la memoria y el ancho de banda del procesador, el núcleo puede ejecutar sólo una

instrucción a la vez y las tareas corren el riesgo que se obstruyan entre sí. Por otra parte, los FPGAs no necesitan de sistemas operativos, los restos de fiabilidad se minimizan con la ejecución paralela y hardware precisado a cada tarea.

Mantenimiento a largo plazo: Los chips FPGAs son reprogramables en campo por lo que no requieren del tiempo y costo que implica rediseñar un ASIC. Un protocolo de comunicación digital tiene especificaciones que podría cambiar con el tiempo lo cual podría causar retos de mantenimiento a las interfaces basadas en ASICs. Al ser reconfigurables, los FPGAs son capaces de lograr las modificaciones necesarias que pudieran presentarse en un futuro y mejorar su funcionalidad sin la necesidad de invertir en tiempo rediseñando el hardware o alterando el diseño de la tarjeta (National Instruments, 2011).

2.4 Distribución normal

También conocida como distribución Gaussiana, es una de las distribuciones más importantes en la estadística ya que describe aproximaciones de fenómenos físicos reales que ocurren en la naturaleza. La gráfica en una distribución normal tiene una forma acampanada por lo que también se les puede conocer como campanas de Gauss, en honor al matemático Karl Friedrich Gauss.

La curva normal depende de dos parámetros estadísticos, la media (μ) y la desviación estándar (σ). Las peculiaridades de la curva normal son (Walpole *et al.*, 2012):

1. La moda ocurre en $x=\mu$ y se ubica en el eje de las abscisas cuyo valor corresponde al punto más alto de la curva normal.
2. La curva es simétrica respecto a su media μ . Es decir, hacia cualquiera de los lados de μ , la imagen es un espejo respecto al eje vertical ubicado este punto.

3. La curva tiene puntos de inflexión en $x=\mu\pm\sigma$. Es cóncava hacia arriba cuando $\mu-\sigma < x < \mu+\sigma$, y es cóncava hacia abajo cuando x se encuentra fuera de este rango.
4. Las variables probabilísticas media, mediana y moda tienen el mismo valor
5. El área total bajo la curva y sobre el eje de las ordenadas es igual a la unidad.
6. Conforme se aleja de la media, la gráfica se aproxima al eje de las abscisas de manera asintótica.
7. Si se trazan perpendiculares a una distancia de tres desviaciones estándar desde la media μ hacia ambos lados, el área bajo la curva delimitada por esas perpendiculares corresponde a un 99.7 por ciento del área total. Si esas perpendiculares se trazan a una distancia de dos desviaciones estándar hacia ambos lados de la media μ , la curva comprenderá aproximadamente un 95 por ciento del área total. Mientras que si se las perpendiculares se reducen a una distancia de una desviación estándar desde la media μ a ambos lados el área total delimitada es del 68 por ciento del área total (Figura 2.11).
8. Las variables μ y σ describen la forma de la curva. Los valores distintos de μ desplazan la curva a través del eje de las abscisas, mientras que los valores de σ determinan el grado de aplanamiento en la curva.

Función de Densidad Distribución Gaussiana

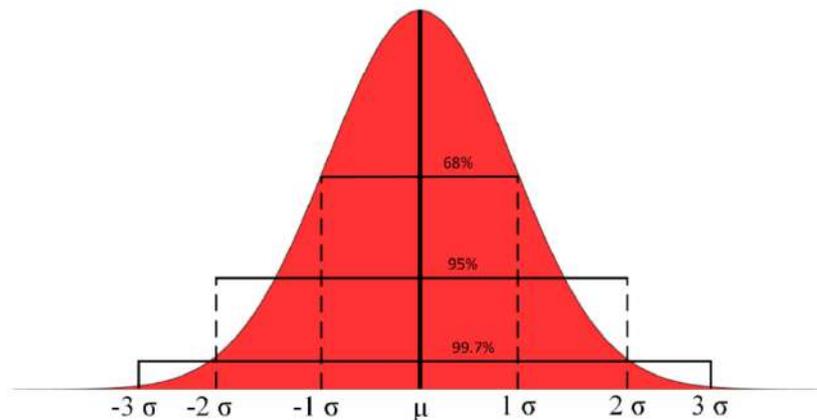


Figura 2.11 Función de distribución Gaussiana.

3. METODOLOGÍA

En este capítulo se describe el diseño del lazo de seguimiento de fase propuesto para la sincronización de las tarjetas DUA 4. Para un mejor entendimiento del comportamiento del sistema, se describen a continuación las señales principales que rigen los eventos internos del sistema:

- ID_{CLOCK} = Señal del oscilador de la tarjeta DUA 4. Es provista por un reloj de cuarzo que trabaja a 48 MHz y posee una tolerancia de +/- 100 PPM.
- CLK_N = Señal de 1 kHz. Es la señal de resolución deseada a la salida del lazo de seguimiento de fase.
- PPS = Señal con alta estabilidad y exactitud que es provista por el módulo GPS. Mantiene un ancho de pulso de 4 ms y posee una frecuencia de 1 Hz.
- CLK_{Sync} = Señal de 1 Hz generada internamente por la tarjeta DUA 4.

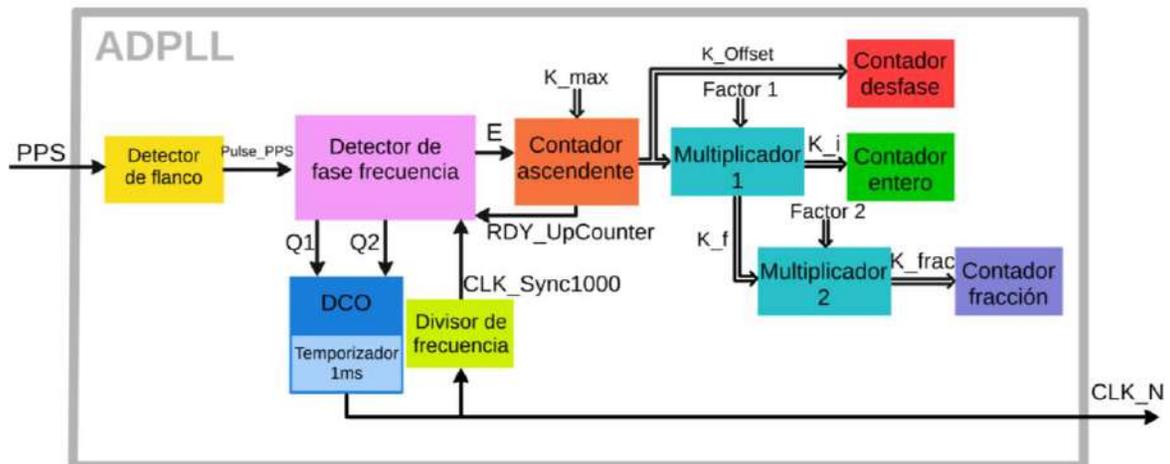


Figura 3.1 Diagrama de ADPLL propuesto.

La Figura 3.1 muestra un diagrama a bloques del diseño. Como entrada se tiene la señal PPS y como salida la señal CLK_N de 1 kHz, la cual hereda las mismas características de estabilidad que ID_{CLOCK} . Este módulo de sincronización está constituido internamente por un detector de flancos ascendentes, un detector de

fase frecuencia, cuatro contadores, dos multiplicadores, un divisor de frecuencia y un oscilador digital controlado tal y como lo muestra la Figura 3.1.

3.1 Detector de flancos

Dentro de la literatura consultada se observa que, en la mayoría de los diseños propuestos los lazos de seguimiento de fase utilizan el área de superposición entre la señal de referencia y la de retroalimentación para calcular el error de fase. No obstante, existen factores ambientales no controlados que pueden alterar la conexión entre la estación satelital y el receptor GPS, haciendo que la señal PPS se pierda durante un periodo de tiempo. Es por esta razón que el sistema debe trabajar a partir de las comparaciones de las transiciones de un estado a otro. El detector de flancos es un módulo capaz de detectar el cambio de estado de bajo a alto; está compuesto por dos *flip-flop* tipo D, una compuerta *AND* y una compuerta *NOT* como se muestra en la Figura 3.2.

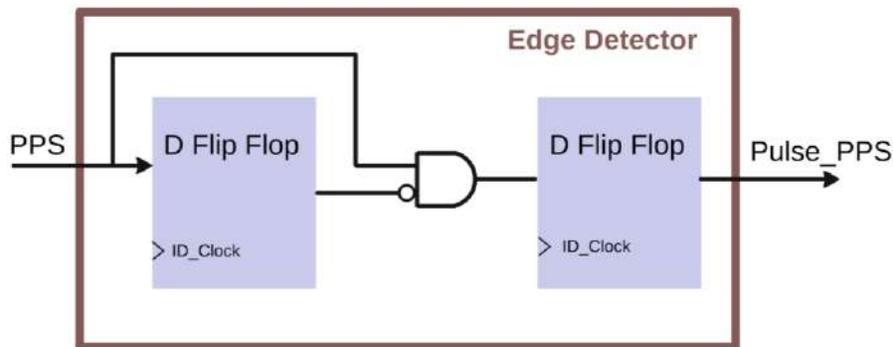


Figura 3.2 Diagrama interno del detector de flancos.

La señal de entrada del detector de flancos es la señal *PPS*, proveniente del receptor GPS y como salida se obtiene la señal *Pulse_{PPS}*. Esta segunda señal permanece la mayor parte en bajo y cambia a estado lógico alto cuando se detecta un flanco ascendente de la señal *PPS*. Su duración en estado alto es inversamente proporcional a la frecuencia del *ID_{clock}*.

3.2 Contadores y cálculos de factores

El módulo contador ascendente se trata de un contador limitado que muestra la cantidad de desfase que existe entre la señal de referencia PPS y la señal de retroalimentación CLK_{Sync} . Se conforma por un contador ascendente que tiene como entrada la señal de activación E y un factor K_{max} ; mientras que a la salida se encuentra la señal $RDY_{UpCounter}$ y el arreglo de señales K_{Offset} . El contador mantiene la cuenta mientras la habilitación, E , se encuentre en estado lógico bajo; en el caso contrario, K_{Offset} aumentará. La señal $RDY_{UpCounter}$ se activará únicamente cuando el valor de la señal de salida, K_{Offset} , sea igual a la entrada K_{max} .

El factor K_{max} representa la relación entre el desfase máximo para una señal de 1 Hz y la frecuencia del oscilador principal de la tarjeta. Se pretende que la señal CLK_{Sync} mantenga las mismas condiciones de estabilidad que el ID_{CLOCK} . Utilizando las ecuaciones 1 y 2, se obtiene que:

$$f_{min} = 0.9999 \text{ Hz}$$

$$f_{max} = 1.0001 \text{ Hz}$$

Lo cual corresponde a una variación del periodo de $\Delta T_{1Hz} = 1.0 \times 10^{-4} \text{ s}$ para ambos casos. La relación entre la diferencia de periodo y el factor K_{max} del contador ascendente está dado mediante la ecuación 5.

$$K_{max} = (\Delta T_{1Hz})(ID_{CLOCK}) \quad (5)$$

$$K_{max} = 4800$$

La señal a la salida del contador ascendente (K_{Offset}) es modificada por el detector de fase frecuencia. En la sección 2.2.1 se describió el principio de operación y el rango de los detectores de fase frecuencia, así como también se

explicó el comportamiento de la señal de salida de este módulo. Para la arquitectura desarrollada en el presente proyecto, el rango de detección se mantiene entre -2π y 2π . Sin embargo, la señal de salida del contador ascendente no se mantiene proporcional dentro del rango mencionado. Existe un comportamiento lineal entre -4.83×10^{-5} rad y 4.83×10^{-5} rad, y fuera de estos valores la señal K_Offset presenta saturación, tal como lo muestra la Figura 3.3

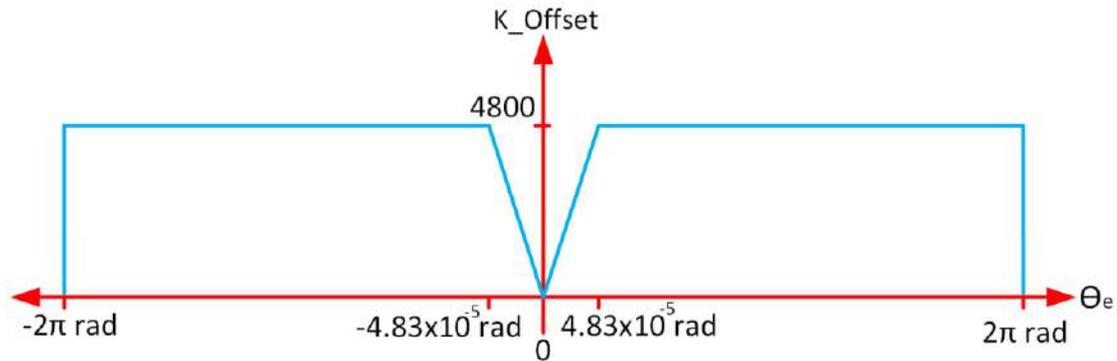


Figura 3.3 Salida del contador ascendente contra el error de fase.

Como se ha mencionado al inicio de este capítulo, es necesario que la señal de 1 KHz obtenida a la salida del lazo de seguimiento de fase mantenga las mismas condiciones de estabilidad que el reloj principal de la tarjeta. Es por lo anterior que la compensación se debe realizar proporcionalmente entre cada uno de los ciclos de la señal CLK_M . En el módulo contador ascendente se calculó la diferencia de tiempo entre periodos para una señal de 1 Hz. Sin embargo, esta cantidad no corresponde al desfase para una señal de 1 ms. Utilizando las ecuaciones 1 y 2 para obtener las frecuencias máximas y mínimas de una señal de 1 KHz y tolerancia de ± 100 PPM se obtiene que:

$$f_{min} = 999.90001 \text{ Hz}$$

$$f_{max} = 1000.1 \text{ Hz}$$

Para ambos casos corresponde a una variación del periodo de $\Delta T_{1Hz} = 1.0 \times 10^{-7}$ s. Haciendo uso a la ecuación 3, se tiene entonces que el factor necesario para la compensación máxima de CLK_N es de:

$$K_{i(\max)} = 4.8 \approx 5$$

Si el valor resultante se truncara a 4 se comprometería la sincronización en la señal de 1 Hz ya que el sistema no compensaría en su totalidad el desfase máximo entre la señal CLK_{Sync} y la señal $Pulse_{PPS}$. Lo anterior resultaría en un error de fase mínimo entre estas señales que se acumularía y aumentaría a lo largo del tiempo. Para corregir este error se realiza un redondeo a 5, pero además se realiza un cálculo del factor necesario para la compensación de la parte fraccionaria.

El valor obtenido por el contador ascendente pasa a un multiplicador, que multiplica al factor K_{Offset} por un valor fijo (*Factor1*) igual a 0.001 o lo que es lo mismo, dividir el valor entre 1000 partes iguales. A la salida de este primer multiplicador se obtienen dos resultados: la parte entera (K_i) y la parte fraccionaria de la operación (K_f). El resultado K_i entra al módulo contador entero, mientras que K_f entra a un segundo multiplicador.

Este segundo módulo de multiplicación tiene como finalidad convertir la parte fraccionaria a un valor entero, por lo que K_f debe ser multiplicado por un valor de 1000 y el resultado, K_{frac} , de esta operación entra al módulo contador fracción. Es importante mencionar que para que la señal CLK_N evite fluctuaciones en su frecuencia, la cantidad K_{frac} calculada deberá irse compensando de uno a uno en cada ciclo de CLK_N hasta alcanzar el valor antes mencionado.

Tabla 3-1 Valores de los factores calculados para el diseño del ADPLL.

Nombre	Tipo	Valor decimal	Formato binario
K_{max}	Fijo	4,800	13.0
K_{Offset}	Variable	0-4,800	13.0
$Factor_1$	Fijo	0.001	0.32
K_i	Variable	0-5	3.0
K_f	Variable	0-0.999	0.32
$Factor_2$	Fijo	1,000	10.0
K_{Frac}	Variable	0-999	10.0

En resumen, la tabla 3-1 muestra un listado de las variables que entran a los módulos de conteo antes mencionados, así como el rango de valores máximos y mínimos que estas variables pueden tomar. Se muestra, además, el tipo de formato binario que cuenta cada valor.

3.3 Oscilador digital controlado (DCO)

Este módulo es de gran importancia en el lazo de seguimiento de fase y es la parte en donde se realiza la compensación en tiempo de la señal a sincronizar. Está compuesto por un temporizador operando a la frecuencia de resolución deseada (1 KHz) y trabaja en conjunto con el detector de fase frecuencia. Posee dos entradas: Q_1 y Q_2 , las cuales modifican directamente la tasa de cambio del temporizador del DCO.

En las estructuras digitales, un temporizador se describe como un contador de ciclos del ID_{CLOCK} y un comparador. El contador se encuentra activo en todo momento y cuando la cuenta actual del contador logre alcanzar la cantidad

necesaria para obtener una señal de 1 ms, un pulso en estado lógico alto (CLK_N) se mantendrá a la salida del *DCO*.

Cuando la señal Q_1 está en alto se presenta un atraso de la señal CLK_{Sync} con referencia a la señal $Pulse_{PPS}$, por lo que la base de tiempo del temporizador deberá disminuirse. Esto se logra aumentando la tasa de cambio mientras Q_1 esté en estado lógico alto. En el caso contrario en el cual se presente un adelanto de la señal CLK_{Sync} con referencia a la señal $Pulse_{PPS}$, la base de tiempo deberá aumentarse, manteniendo la cuenta actual del temporizador mientras Q_2 se mantenga en estado lógico alto. Este principio de funcionamiento puede verse en la tabla 3-2

Tabla 3-2 Tabla de verdad del DCO.

Entradas		Cuenta (n+1)
Q1	Q2	
0	0	Cuenta(n)+1
0	1	Cuenta(n)
1	0	Cuenta(n)+2

3.4 Divisor de frecuencia

En este módulo se divide la frecuencia de la señal obtenida por el *DCO*. CLK_N es una señal con una frecuencia de 1 KHz, por lo que no puede ser utilizada para la comparación en el detector fase frecuencia. Un escalamiento de su frecuencia debe ser implementado antes de retroalimentar la señal al comparador de fase frecuencia.

A la entrada del módulo se encuentra la señal de CLK_N (1KHz) y a la salida se obtiene la señal CLK_{Sync} (1Hz). El factor de escalamiento se obtiene mediante la ecuación 6.

$$N = \frac{f_{in}}{f_{out}} \quad (6)$$

Por lo que para la aplicación deseada $N = 1000$.

3.5 Detector de fase frecuencia

Es la parte del sistema que coordina la interacción de los módulos internos y se describe mediante una máquina de estados finita. El principio de operación del detector de fase frecuencia se muestra en la Figura 3.4 mediante un diagrama de flujo. El sistema se encuentra en estado de espera la mayoría del tiempo (estado de sincronía) y las banderas Q_1 y Q_2 se desactivan en este estado.

El ciclo de compensación se inicia únicamente cuando el detector de fase frecuencia detecte un desajuste entre los flancos ascendentes de las señales CLK_{Sync} y $Pulse_{PPS}$. En este instante se inicia el conteo de desfase; dicho conteo se interrumpe únicamente en dos casos: que la cuenta del contador ascendente alcance la cuenta K_{max} y la segunda, que $Pulse_{PPS}$ sea detectado. Las señales Q_1 y Q_2 continúan desactivadas durante este periodo de tiempo.

Como segundo paso se realiza la compensación de la parte fraccionaria correspondiente a un CLK_N . Los módulos contador ascendente y contador fracción aumentan la cuenta en uno y almacenan este nuevo valor. Por otra parte, contador entero se encuentra desactivado y las banderas Q_1 y Q_2 se activan o desactivan según sea el caso. Si CLK_{Sync} está en adelanto a $Pulse_{PPS}$ Q_1 toma el estado lógico bajo y Q_2 se mantiene en estado lógico alto. En caso contrario, en el que CLK_{Sync} esté en atraso con respecto a la señal $Pulse_{PPS}$ Q_1 se activará mientras que Q_2 se mantendrá en bajo.

El tercer paso consiste en realizar la compensación de la parte entera para un ciclo de la señal CLK_N . Los módulos contador entero y contador desfase se activan mientras que contador fracción sigue manteniendo la cuenta actual. Las señales Q_1 y Q_2 toman el mismo valor que en el paso dos. Este proceso de

compensación para la parte entera termina cuando el valor actual en el contador entero es equivalente a K_i .

En el cuarto paso se encuentra la generación del resto de la señal CLK_N . El módulo contador entero se desactiva, mientras los módulos contador fracción y contador desfase mantienen la cuenta y ambas banderas, Q_1 y Q_2 , se desactivan.

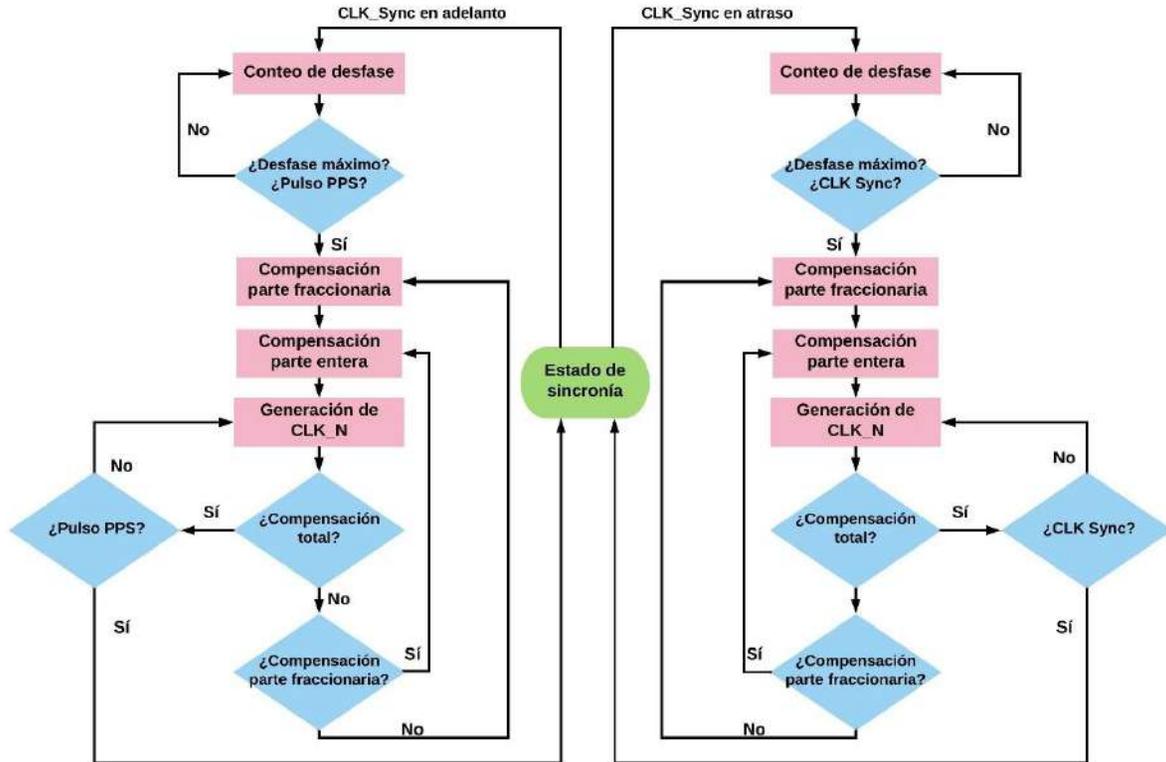


Figura 3.4 Diagrama de flujo del detector fase frecuencia.

Una vez terminado el cuarto paso, se realizan las comparaciones. Si la cuenta del contador desfase coincide con factor K_{max} el sistema se queda ciclado en espera a la señal $Pulse_{PPS}$, generando la señal CLK_N sin realizar ningún tipo de compensación. En el caso contrario en el que no se terminase con el proceso de compensación, una nueva comparación se hace presente. Si no se ha realizado la compensación de la parte fraccionaria, es decir, que la cuenta de contador fracción no corresponda al valor de K_{Offset} , se repite el proceso desde el paso dos. De otra manera, sólo los pasos tres y cuatro son ciclados. El sistema regresa al estado de

sincronía únicamente cuando se ha terminado de realizar la compensación total y la señal $Pulse_{PPS}$ es detectada.

En la sección 3.3 se calculó el valor necesario para la compensación de la señal de 1 kHz en el caso de que sucediera el desfase máximo entre las señales CLK_{Sync} y $Pulse_{PPS}$. Como se mencionó en dicha sección, el valor para la compensación de la parte entera es cuatro y durante los primeros 800 ciclos de CLK_N se presentaría una compensación de la parte fraccionaria. En resumen, en los primeros 800 ciclos se compensaría un valor de 5 mientras que los 200 ciclos restantes de CLK_N se compensaría lo correspondiente a un factor de 4. Este pequeño aumento en la compensación de la parte entera genera que el valor de la tolerancia aumente para la señal de 1 kHz. En la ecuación 7 se muestra la relación entre la tolerancia y el aumento de los factores de compensación.

$$PPM = \frac{1 \times 10^6 (K_{i(max)} + 1)}{(f_{IDCLOCK})(T_{CLK_N})} \quad (7)$$

El nuevo valor de la tolerancia aplicada para CLK_N es de 104.16 PPM. Este aumento del 4.16% en la tolerancia se presentará solamente en aquellos casos en cuyo factor K_{Offset} sea mayor o igual a 4,001. En caso contrario, la tolerancia permanece dentro del rango especificado al inicio del diseño, el cual es de +/- 100 PPM.

3.6 Rangos de frecuencias de operación

Es importante conocer el rango de frecuencias a las cuales opera el sistema desarrollado. Debido a que el lazo de seguimiento de fase trabaja como un sintetizador de frecuencias, el análisis de los rangos de frecuencias se debe dividir en dos: para el primero, se toma en como frecuencia central (ω_0) la señal a la cual se compara la entrada en el lazo (1 Hz), mientras que en el segundo análisis la frecuencia central es determinada por la frecuencia natural de operación del DCO (1 kHz).

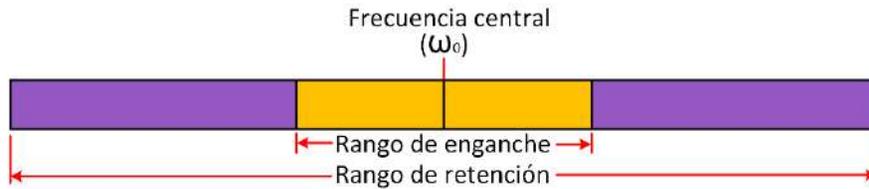


Figura 3.5 Rangos de operación en el PLL.

La frecuencia central se refiere a la frecuencia ideal de oscilación del sistema. Como se mencionó con anterioridad, se tienen dos frecuencias centrales: a la salida del DCO y a la salida del divisor de frecuencia, aunque la de mayor interés en el presente trabajo es aquella de 1 kHz y es provista por el oscilador digital controlado. Cuando el lazo de seguimiento de fase se mantiene operando a la frecuencia central, quiere decir que el error de fase entre la señal generada y la señal de referencia es cero.

El rango de enganche se refiere al conjunto de frecuencias en las cuales el lazo de seguimiento de fase se puede volver a enganchar antes de haber transcurrido un periodo de la señal de salida. Para la señal de 1 KHz, el rango de enganche está entre los 999.89 Hz y 1000.10 Hz, mientras que para la señal de 1 Hz este rango fluctúa entre 0.9999 Hz y 1.0001 Hz. A pesar de las variaciones de frecuencias en ambos casos son mínimas, esto permite el enganche de la señal generada con la señal de referencia manteniendo un alto grado de exactitud.

El rango de retención es el conjunto de frecuencias para las cuales el PLL puede seguir a la referencia y logrará mantenerse enganchado, aunque este proceso puede tomar tiempo, ya que las variaciones de frecuencia que se realizan son pequeñas. Este rango representa los límites estáticos de estabilidad en el sistema y están directamente relacionados al detector de fase implementado. Para la señal de 1 kHz el rango de retención se encuentra entre los valores de +/- 1 kHz a partir de la frecuencia central. Para la señal de 1 Hz este rango se mantiene entre +/- 1 Hz, medido desde la frecuencia central.

4. EXPERIMENTACIÓN Y RESULTADOS

En este capítulo se muestra y describe el experimento realizado, así como también se detallan los resultados obtenidos. En la Figura 4.1 se muestra un diagrama de la puesta del experimento desarrollado para la validación del lazo de seguimiento de fase diseñado. En una primera prueba, dos tarjetas DUA 4 adquieren una misma señal senoidal de 100 Hz. Ambas tarjetas muestrean a una frecuencia de 1 kHz sin poseer ningún elemento de sincronización. Luego de diez minutos de adquisición continua, los datos obtenidos por ambas tarjetas son comparadas mediante el software MATLAB.

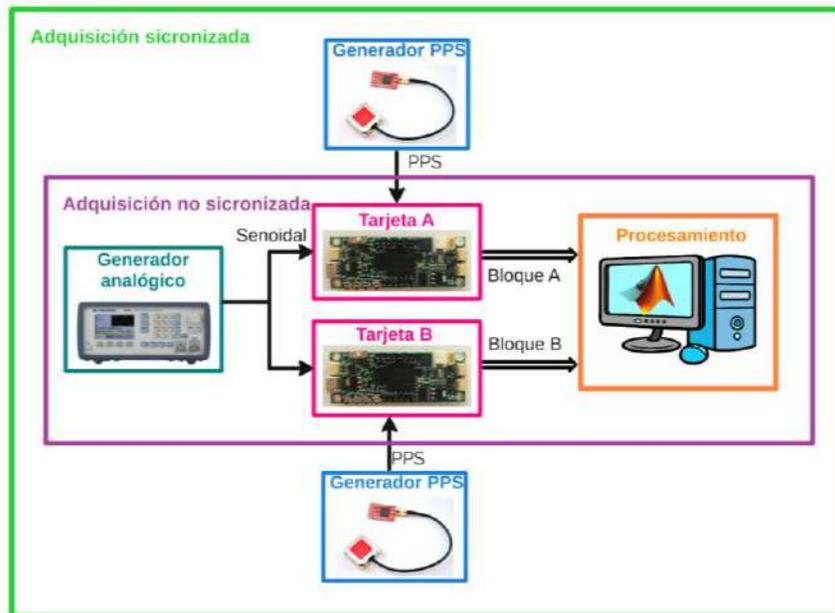


Figura 4.1 Esquema general de la puesta del experimento.

La segunda puesta resultó ser similar a la anterior, con la diferencia de que esta vez se implementó en ambas tarjetas el lazo de seguimiento diseñado. Se requirió, además, de una antena y un receptor GPS, los cuales proveían de la señal de referencia a la cual el ADPLL debía engancharse.

4.1 Adquisición no sincronizada

En la Figura 4.2 se muestran las señales adquiridas normalizadas de ambas tarjetas al inicio de la adquisición. El desfase entre ambas es imperceptible; el error de desfase acumulado al inicio de la adquisición es cero, por lo que se puede concluir que la señal A y la señal B están sincronizadas tanto en fase como en frecuencia.

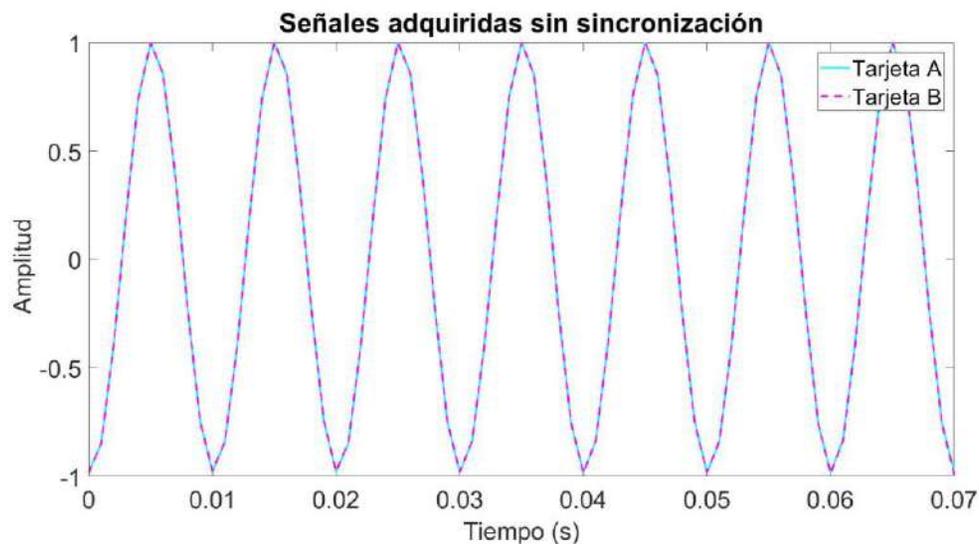


Figura 4.2 Señales adquiridas sin sincronización al inicio de la adquisición.

La Figura 4.3 muestra el desfase entre ambas señales luego de 30 segundos de adquisición continua. Aunque el error de fase es poco, resulta ser evidente que el error hasta ahora acumulado es mayor que al inicio de la adquisición. Las señales parecen conservar su frecuencia, su amplitud, más no el ángulo de fase entre ellas.

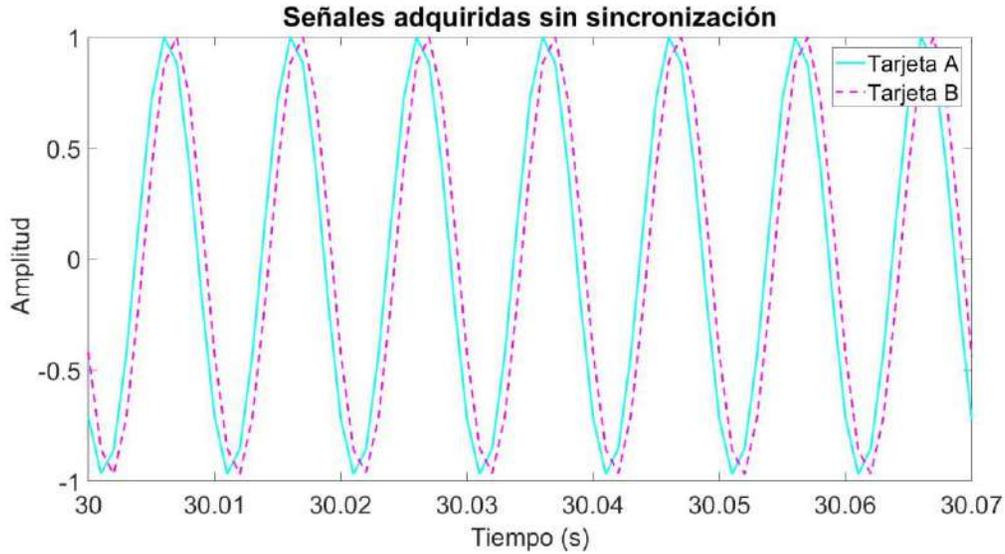


Figura 4.3 Señales adquiridas sin sincronización después de 30 s de adquisición continua.

Teóricamente, el peor escenario que puede existir durante la adquisición es que el oscilador de ambas tarjetas trabaje justo en los límites de estabilidad, es decir, que la tarjeta A adquiriera sólo a frecuencias máximas, mientras que la tarjeta B se encuentre trabajando sólo a frecuencias mínimas. Dicho en otras palabras, que la frecuencia de muestreo en el dispositivo A es de 1 kHz +100 PPM y en la tarjeta B es de 1 kHz -100 PPM. En este caso, la diferencia entre el periodo nominal y el real sería de:

$$\Delta T_A = 1.0 \times 10^{-7} \text{ s}$$

$$\Delta T_B = 1.0 \times 10^{-7} \text{ s}$$

Por lo que la diferencia total en tiempo entre la señal de muestreo de la tarjeta A y la tarjeta B es de:

$$\Delta T_{max} = \Delta T_A + \Delta T_B = 2.0 \times 10^{-7} \text{ s}$$

Se sabe que la señal analógica adquirida posee un periodo de 0.01 s, debido a lo cual un ángulo de 2π corresponde a un periodo de tiempo de $5 \times 10^{-3} s$. La ecuación 8 muestra la relación que existe entre el ángulo de fase de las señales, la variación de periodo en el oscilador y el tiempo de adquisición transcurrido:

$$t_{min} = \frac{T(\varphi)}{4\pi(\Delta T_{max})} (T_s) \quad (8)$$

donde:

t_{min} = tiempo de muestreo mínimo transcurrido en segundos

φ = ángulo de fase entre señales adquiridas

T = periodo de la señal muestreada

$\Delta T_{máx}$ = diferencia de tiempo máximo en la señal de muestreo

ΔT_s = periodo de muestreo

Para el caso antes mencionado, en el que existe un desfase de 180° entre las señales adquiridas A y B, el tiempo mínimo de adquisición es de 25 segundos. Esto quiere decir que en la realidad no se puede dar un desfase igual o mayor a 2π en menos de 25 segundos. Si esto sucediera, indicaría que los osciladores de las tarjetas no se encuentran operando dentro de la tolerancia señalada. La Figura 4.4 muestra un ángulo de fase entre señales equivalente a 2π al cabo de 250 segundos. Esto implica que ambas tarjetas se encuentran trabajando al menos diez veces por debajo de los límites especificados por el proveedor.

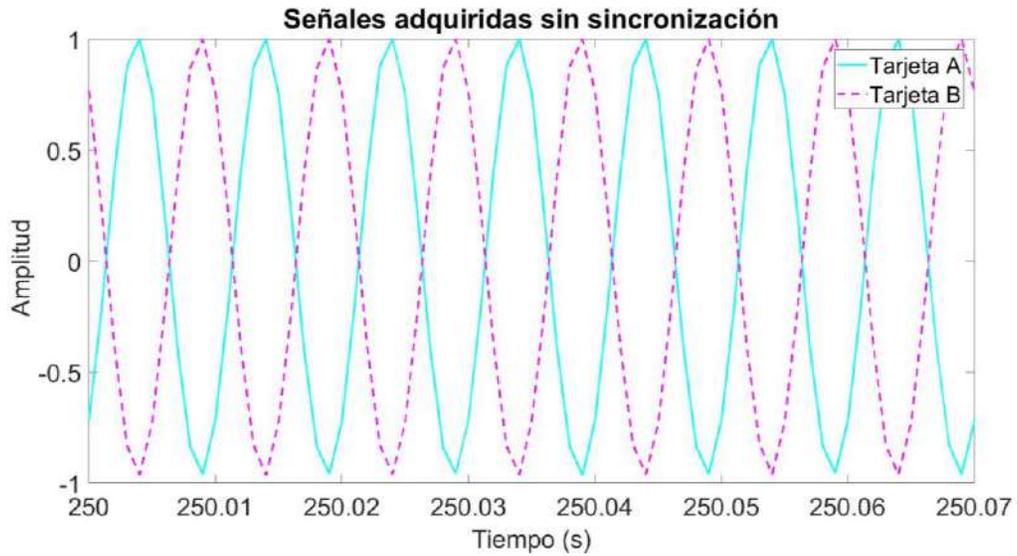


Figura 4.4 Señales adquiridas sin sincronización luego de 250 s de adquisición continua.

No obstante, el error sigue acumulándose a lo largo del tiempo. Luego de 496 segundos de adquisición ininterrumpida el desfase entre señales es tal, que las señales vuelven a sobreponerse una con otra tal y como se observa en la Figura 4.5, esto se debe a que existe ahora un desfase de 360° .

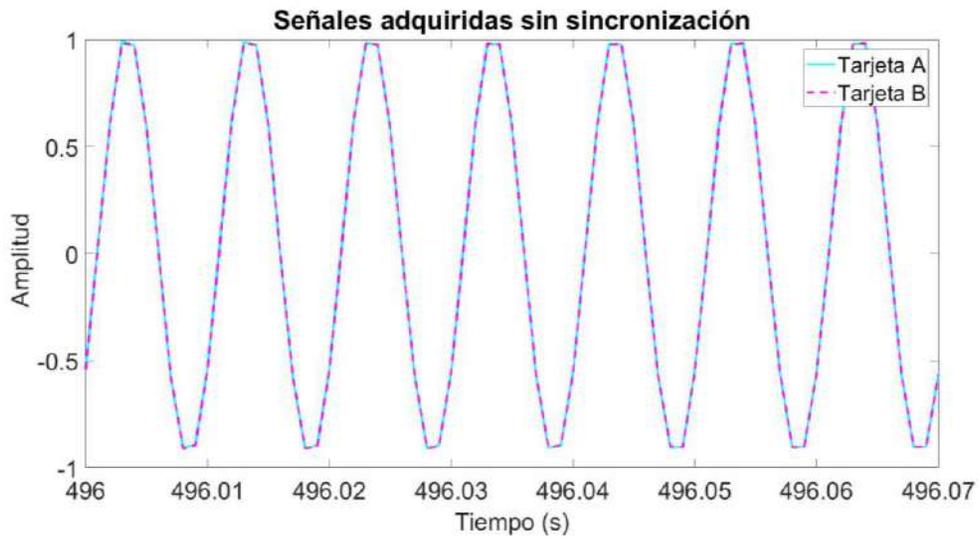


Figura 4.5 Señales adquiridas sin sincronización luego de 496 s de adquisición continua.

Posteriormente se calculó el error de amplitudes de ambas señales y de estos resultados se obtuvieron las envolventes superior e inferior como lo muestra la Figura 4.6. El valor máximo calculado en el eje de las ordenadas (Error) fue de 2, mientras que el valor mínimo fue de -2. Los valores máximos y mínimos ocurren cuando el ángulo de fase entre señales es de 2π .

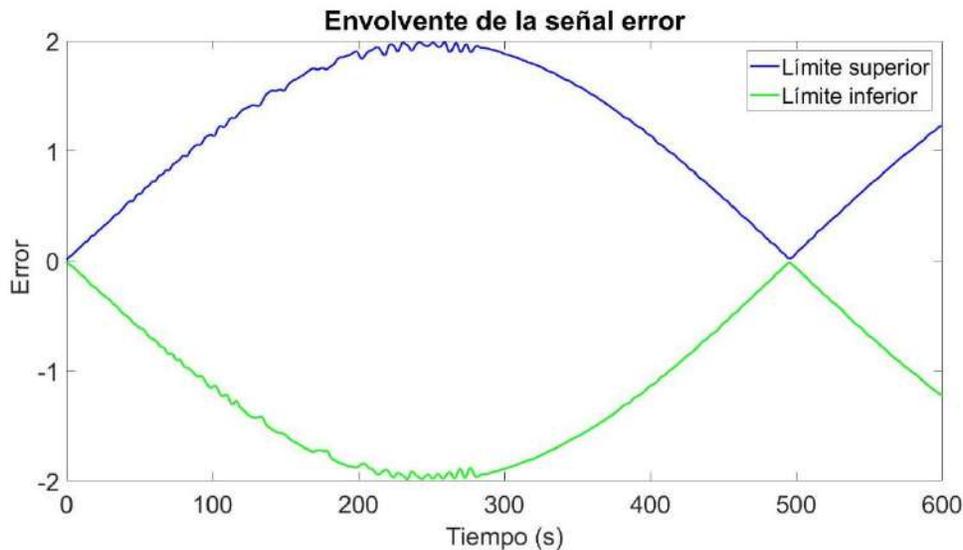


Figura 4.6 Señales envolventes del error sin sincronización

4.2 Adquisición sincronizada

4.2.1 Adquisición sincronizada continua

De manera similar a la puesta anterior, se obtuvieron las señales en ambas tarjetas a una frecuencia de 1 kHz, con la diferencia de que en esta ocasión el periodo de muestreo se encontraba regido mediante el módulo de sincronización. La Figura 4.7 muestra los datos obtenidos por las dos tarjetas al inicio de la adquisición.

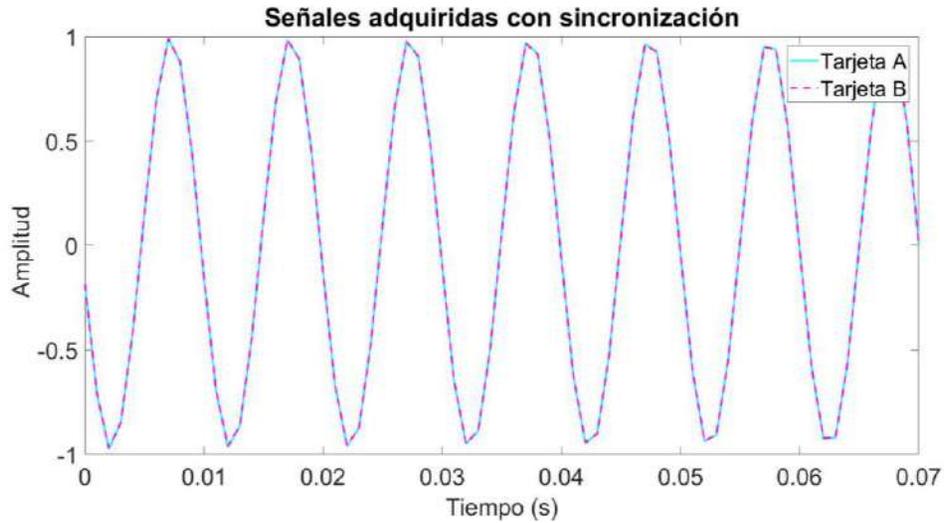


Figura 4.7 Señales adquiridas con sincronización al inicio de la adquisición.

Como se puede observar, ambas señales se encuentran en sincronía al inicio de la prueba y el error de fase acumulado hasta ahora es muy cercano a cero. A diferencia a la adquisición anterior, en este caso se espera que el comportamiento de las señales mostrado en la figura anterior se mantenga a lo largo del tiempo. En la Figura 4.8 se muestra que luego de 250 segundos de adquisición continua no existe ningún incremento en el ángulo de fase entre ambas señales.

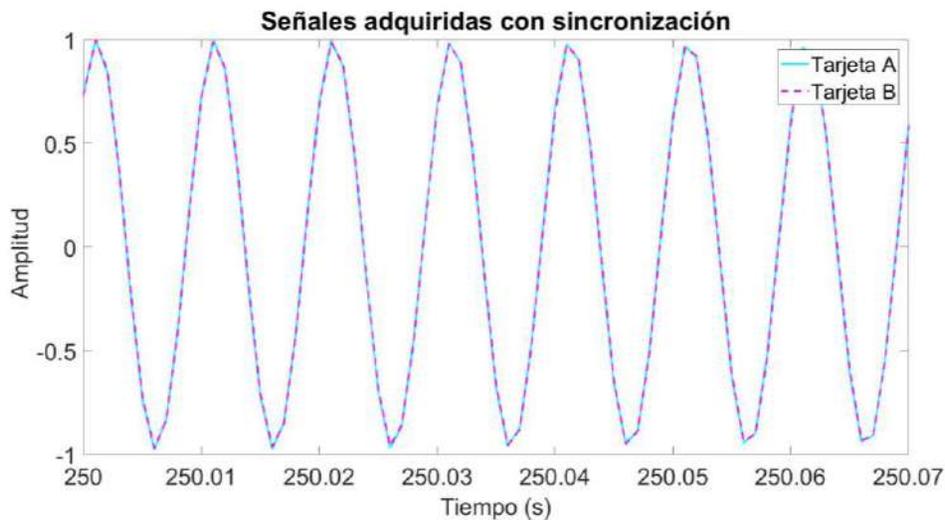


Figura 4.8 Señales adquiridas con sincronización luego de 250 s de adquisición continua.

Al cabo de 496 segundos, el comportamiento anterior se mantiene constante. Hasta ahora el desfase es casi nulo y las señales se siguen observando superpuestas (Figura 4.9)

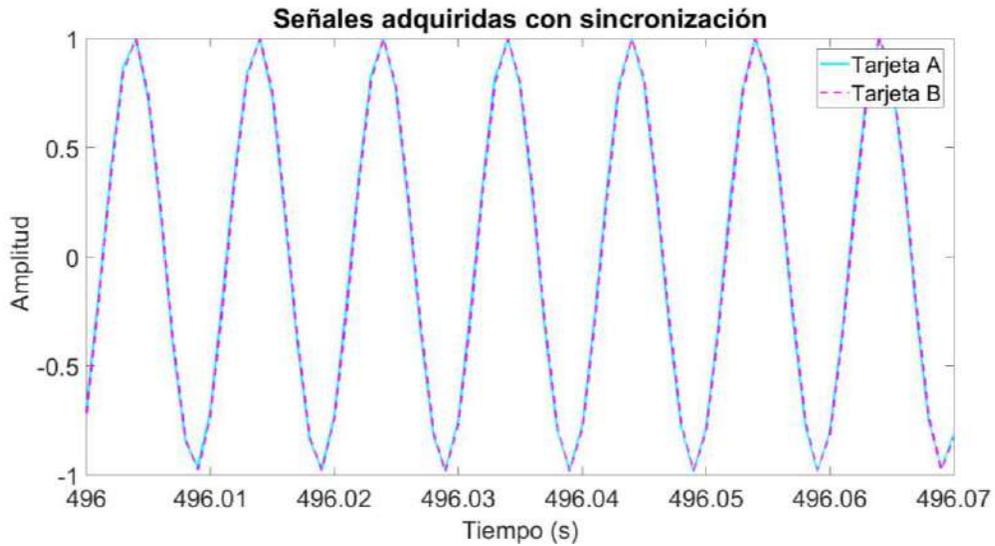


Figura 4.9 Señales adquiridas con sincronización luego 496 s de adquisición continua.

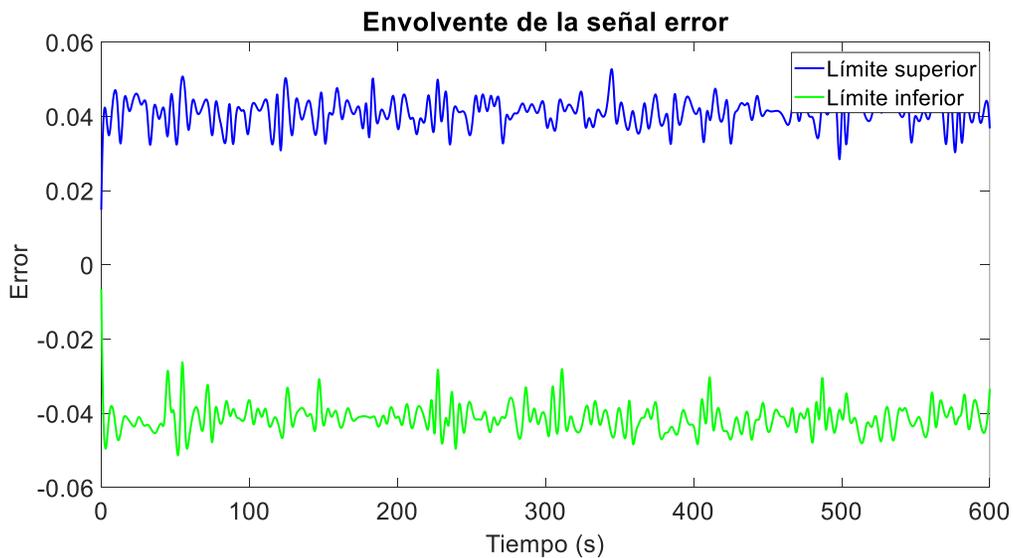


Figura 4.10 Señales envolventes del error con sincronización

De igual forma que en la sección anterior, se calcularon las envolventes superior e inferior de la señal del error. Aunque en los primeros instantes de adquisición la pendiente de ambas envolventes crece súbitamente, se logra estabilizar en un valor cercano a 0.04 y -0.04 manteniéndose con pequeñas variaciones a lo largo del tiempo. En la Figura 4.10 se ilustra el comportamiento mencionado.

4.2.2 Adquisición sincronizada con interrupciones

Con el fin de conocer el comportamiento ante alguna perturbación de la señal de referencia, se realizaron pruebas en las cuales se interrumpió la señal PPS provista por el dispositivo GPS. Existieron tres escenarios, en el primero la señal PPS se interrumpió durante un segundo, en el segundo escenario se interrumpió durante treinta segundos y para el tercero la señal se interrumpió durante un minuto completo. Todas la interrupciones comenzaron un minuto después de comenzar la adquisición y las envolventes del error se pueden observar en las Figuras 4.11, 4.12 y 4.13, respectivamente.

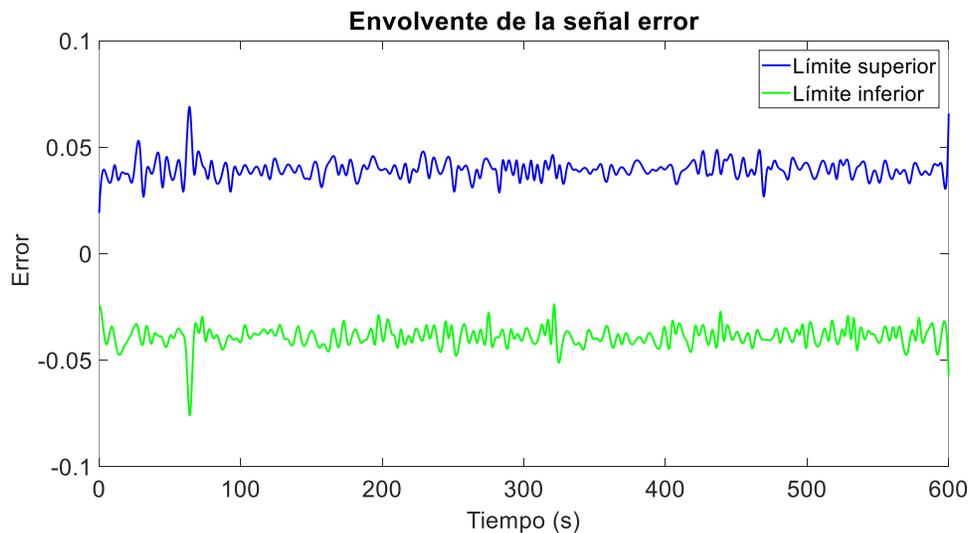


Figura 4.11 Envolventes del error con interrupción de 1 segundo.

Para la prueba de interrupción de 1 segundo el aumento en la magnitud de las envolventes es casi imperceptible con respecto a las envolventes en la adquisición continua. Cuando la referencia es interrumpida este periodo de tiempo, el comportamiento del lazo de seguimiento de fase no se ve afectado.

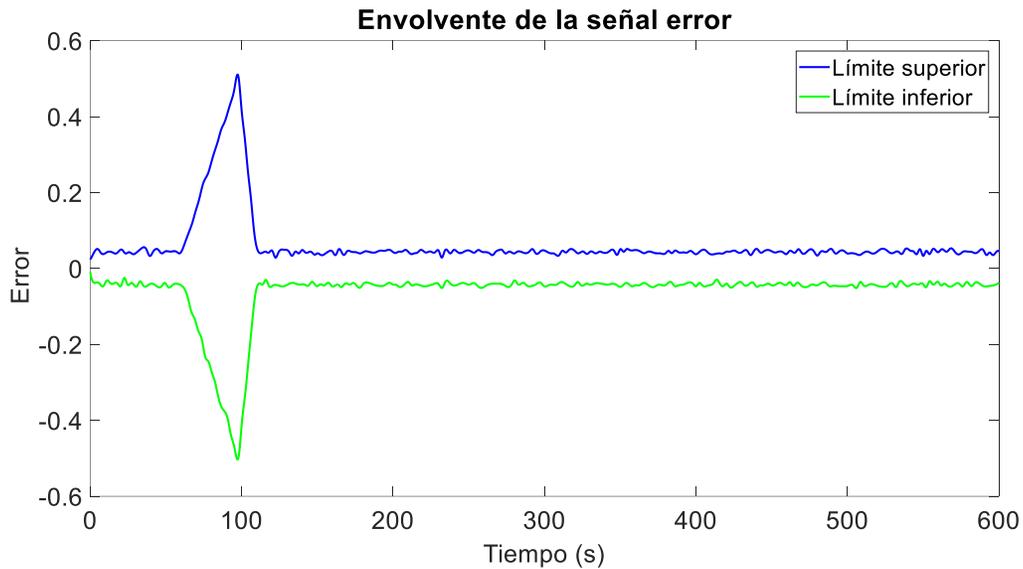


Figura 4.12 Envolventes del error con interrupción de 30 segundos.

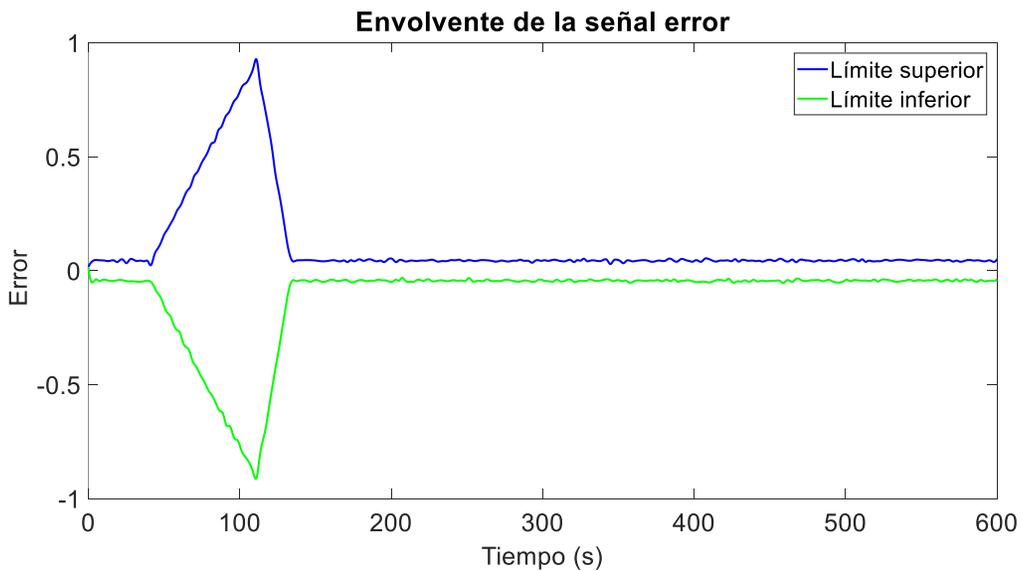


Figura 4.13 Envolventes del error con interrupción de 60 segundos.

En la interrupción de 30 segundos también se observa un aumento en la amplitud de las envolventes del error cuando se presenta la interrupción, la magnitud de la señal del error incrementa de 0.04 a 0.5. Cuando la señal de referencia vuelve a presentarse luego de este periodo de tiempo inactivo, la magnitud en las envolventes regresa a un rango entre -0.04 y 0.04, tal y como se presenta en la adquisición continua.

La gráfica de la adquisición con interrupciones de 60 segundos presenta el mismo comportamiento anterior, con la diferencia que la magnitud en las envolventes vuelve a aumentar. Entre los segundos 60 y 120 de la adquisición se presentó la interrupción. Luego de este período de tiempo la referencia vuelve a entrar al sistema, haciendo que las magnitudes de dichas envolventes vuelvan a reducirse hasta que se encuentran en el mismo rango que al inicio de la prueba.

4.3 Medición de error de fase

Por otra parte, se realizó una prueba para conocer el comportamiento del error de fase de la señal de un milisegundo generada y comprobar si este error de fase es corregido una vez implementado el módulo de sincronización en el sistema. Para este propósito se establecieron dos escenarios: uno que no contara ningún tipo de sincronización y otro en el cual se integrara el ADPLL desarrollado, cuyo principio de operación es el mismo. En ambos escenarios se cuenta con una señal de periodo de un milisegundo, la cual se conforma de pulsos con duración de 20.83 ns, es decir, no se trata una señal cuadrada. Es por esto que analizar el error de fase en esta señal resulta complicado, debido a su poca duración en estado alto.

En virtud a lo anterior se optó por generar una señal que cambiase de nivel lógico en cada flanco ascendente de la señal de un milisegundo. Es decir, se obtuvo una señal cuadrada, cuyo ciclo en alto posee la misma duración que su ciclo en bajo. Esto ocasiona que dicha señal cuadrada a analizar posea un periodo de dos milisegundos.

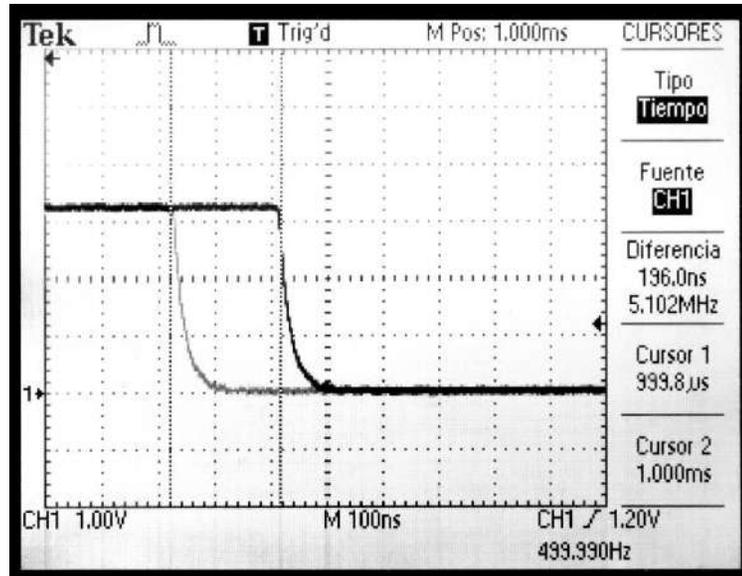


Figura 4.14 Medición de jitter con osciloscopio para señal sin sincronización.

La Figura 4.14 muestra el error de fase presente en la señal con periodo de dos milisegundos generada sin utilizar ningún elemento de sincronización. En ella se observa que la señal presenta un error de fase cuya amplitud es de 196 ns, el cual representa el $9.8 \times 10^{-3} \%$ del periodo total.

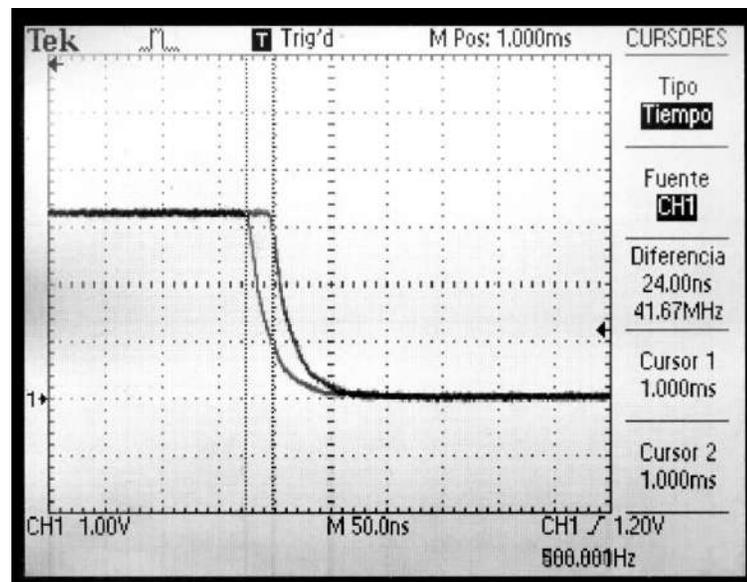


Figura 4.15 Medición de jitter con osciloscopio para señal con sincronización.

Por otra parte, en la Figura 4.15 se observa el error de fase que presenta la señal de dos milisegundos una vez que el módulo de sincronización es implementado. Se puede observar una disminución en la amplitud del jitter. En esta ocasión el error de fase presenta el $1.2 \times 10^{-3} \%$ del periodo total de la señal, es decir, la amplitud máxima del jitter es de 24 ns. La señal generada mediante el módulo de sincronización redujo la amplitud del jitter al menos ocho veces.

4.4 Análisis de resultados

Con la finalidad de garantizar resultados confiables, ambas puestas del experimento (adquisición de datos con y sin sincronización) se repitieron 20 veces. De cada una de estas pruebas se obtuvo el error restando la amplitud de la señal adquirida por la tarjeta A con la tarjeta B. Posterior a ello se obtuvieron las envolventes superior e inferior de la señal del error. De dichas envolventes se calcularon los parámetros estadísticos de desviación, valor RMS y promedio.

Fue necesario el cálculo de estos parámetros estadísticos en las señales envolventes y no directamente en la señal del error debido a que los cambios de signo en la señal del error no representan un adelanto o atraso de las señales A y B, sino que indican un cambio de ciclo positivo a negativo o viceversa en la señal adquirida. Por último, se calculó el valor pico-pico de la señal error. Los valores mostrados en la tabla 4-1 representan el resultado de los parámetros antes mencionados luego de 20 pruebas analizadas.

Tabla 4-1 Parámetros estadísticos calculados.

Parámetro	Resultados con sincronización		Resultados con sincronización	
	Envolvente superior	Envolvente inferior	Envolvente superior	Envolvente inferior
Promedio	1.1390	-1.1377	0.0443	-0.0438
RMS	1.2907	1.2899	0.0447	0.0441
Desviación	0.5922	0.05922	0.0038	0.0038
Valor pico-pico señal error	3.9562		0.1130	

Estudiando los valores presentados en la tabla anterior, resulta evidente que todos los parámetros de las pruebas con sincronización son menores a aquellos que se presentan en las pruebas sin sincronización resaltando así el aporte del diseño propuesto.

Los valores de envolventes nos permiten conocer el ángulo de fase entre las señales A y B. La ecuación 9 muestra la relación entre el valor de las envolventes obtenidas y la diferencia de periodo de la señal de muestreo.

$$\Delta T_s = \frac{T|env|}{4A_0} \quad (9)$$

donde

T = periodo de la señal muestreada

env =valor de la envolvente

A_0 =Valor de la amplitud cero pico de la señal muestreada

Para el cálculo de la diferencia del periodo utilizando la envolvente superior se obtuvo un valor $\Delta T_s = 1.1075 \times 10^{-4}$ s, mientras que para la envolvente inferior se calculó un valor de $\Delta T_s = 1.095 \times 10^{-4}$ s. En ambos casos se presenta una diferencia de periodo mayor al calculado, el cual es de $\Delta T_s = 1.04 \times 10^{-7}$. A pesar de ser un aumento considerable en el periodo de muestreo, estos valores representan una diferencia máxima del 11.075% entre el periodo de muestreo ideal y el real.

En los resultados sin sincronización se cuenta con un alto valor en la desviación estándar y en el valor pico-pico de la señal de error, lo cual indica que las muestras obtenidas en estas pruebas sin sincronización cuentan con un alto grado de variación entre una muestra y otra. Estos resultados indican también que el promedio de las envolventes no es cercano a cero, por lo que cuando no hay sincronización no existe exactitud en las muestras obtenidas. En la Figura 4.16 se pueden apreciar las funciones de distribución normal obtenidas a partir del promedio y desviación estándar de las envolventes superior e inferior. En ambos casos se muestra un bajo grado de concentración alrededor del promedio. Estas curvas permiten concluir que antes de integrar el sistema de sincronización la adquisición de datos carece no sólo de exactitud, sino también de precisión entre sus muestras.

Por otra parte, los valores de la desviación estándar y el valor pico-pico de la señal de error disminuyen cuando se realizaron las pruebas con el sistema de sincronización implementado. En la Figura 4.16 también se muestran las funciones de distribución normal cuando existe sincronización en la adquisición de datos. Los centros de ambas curvas se aproximan a cero y la concentración de los datos se da alrededor de los valores de la media, asegurando así el 99.7% de efectividad en los datos. Para la adquisición con sincronización se cuenta con un alto grado de precisión y de exactitud.

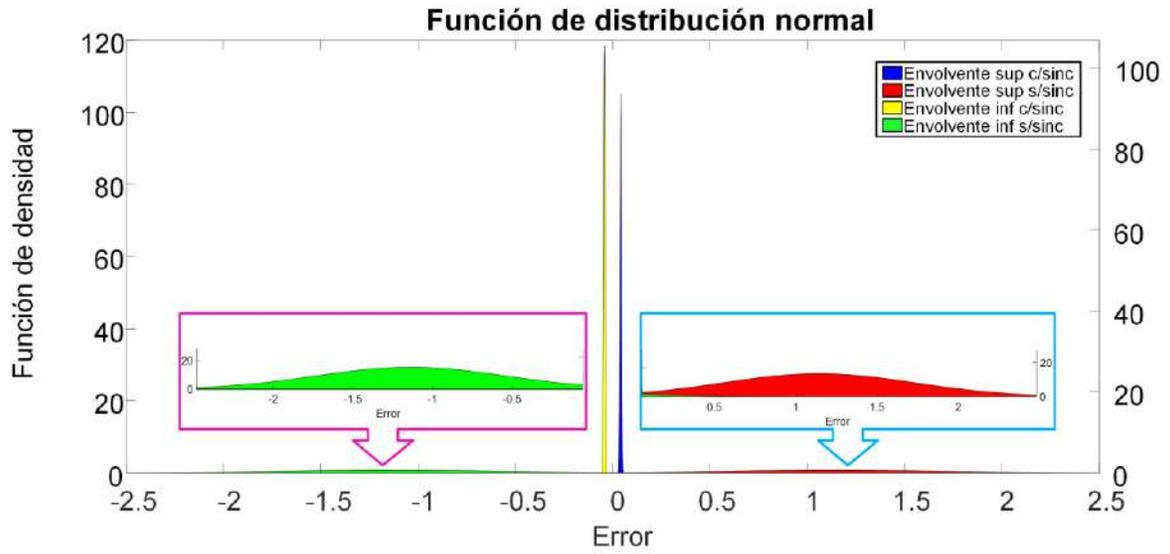


Figura 4.16 Funciones de distribución normal obtenidas de envolventes del error en las adquisiciones con y sin sincronización.

5. CONCLUSIONES

El dispositivo GPS fue configurado mediante mensajes basados en el protocolo de comunicación UART con la finalidad de obtener una señal de un pulso cada segundo (PPS) que tiene baja variación de frecuencia y que sirvió como señal de referencia para el sistema de sincronización. Esta señal mantiene relación con el tiempo universal coordinado.

Se diseñó e implementó un lazo de seguimiento de fase completamente digital que trabaja a partir de las comparaciones de los flancos ascendentes de las señales a sincronizar; por lo que, si el GPS perdiera la conexión satelital, el ADPLL no realiza compensaciones innecesarias que podrían afectar la estabilidad de la señal CLK_N generada a la salida. La señal de 1 Hz obtenida a la salida del lazo de seguimiento de fase presentó poca deriva en el tiempo y mantiene un bajo error de fase incluso en largos periodos de tiempo. Por su parte, se demostró que no era posible generar una señal de 1 kHz que mantuviera las mismas características de estabilidad que el oscilador de la tarjeta DUA 4 debido al error de redondeo que se presentó en los cálculos de los factores para el temporizador que realizaba la compensación en dicha señal. El valor del PPM tuvo que ser ajustado, con un aumento del 4.16 %. Si se deseara obtener una señal de 1 kHz con una tolerancia de ± 100 PPM, se tendría que utilizar un oscilador cuya frecuencia especificada fuera múltiplo a $\frac{1}{\Delta T_{CLK_N}}$. Sin embargo, se logró que la compensación total del error se realiza equitativamente en cada uno de los ciclos de CLK_N , logrando así que la tolerancia en cada ciclo de esta señal se mantuviera constante.

Se desarrolló el convertidor analógico digital para luego ser implementado en conjunto con el lazo de seguimiento de fase. El módulo de sincronización proporcionaba la frecuencia a la cual el convertidor debía operar para adquirir la señal de interés.

Una vez almacenados los datos, se realizó el procesamiento. Este consistió en la normalización de las señales obtenidas por ambas tarjetas, para posteriormente obtener el error entre ambas señales. De la señal de error se obtuvieron las envolventes superior e inferior, así como los valores estadísticos permitieron validar el comportamiento del sistema desarrollado.

Se comprobó que el error calculado de una señal adquirida mediante un módulo con sincronización mejora notablemente comparado a una adquisición sin la implementación del módulo de sincronización. Gracias al cálculo de las funciones de distribución normal se pudo concluir que este diseño no sólo se trata de un sistema preciso, sino también exacto.

Mediante las pruebas con interrupciones se comprobó que si bien el sistema de sincronización propuesto no logra mantener la sincronización aun cuando la señal de referencia se haya perdido, el sistema no induce compensaciones innecesarias que afectan la estabilidad del sistema. El error de fase entre señales aumentó durante las interrupciones, pero éste fue reducido una vez que la referencia fue devuelta al sistema.

La medición del error de fase no se logró en la señal de 1 kHz ya que ésta no es una señal cuadrada, sino que constó de pulsos con duración de 20.83 ns cada uno. Se tuvo entonces que medir el error de fase en una señal cuadrada de periodo de 2 ms generada a partir de la señal de 1 KHz. Fue demostrado que mediante la implementación del sistema de sincronización se logró reducir 172 ns la amplitud del error de fase en dicha señal. Dicho en otras palabras, se redujo el jitter en un 87.76 %.

Pese a que los valores reales indican un aumento de la tolerancia calculada para una señal de 1 kHz, este aumento puede no ser atribuido en su totalidad al diseño propuesto. Existen factores del entorno que agregan errores en la adquisición de los datos, como los es la frecuencia del generador de la señal analógica a analizar. Dicho generador se encontraba oscilando entre un rango de frecuencias de 99.8 Hz y 100.15 Hz durante las pruebas realizadas.

6. BIBLIOGRAFÍA

- Abracon. (03 de 06 de 2014). *Digi-Key Electronics*. Obtenido de <https://www.digikey.com.mx/product-detail/es/abracon-llc/ASE-48.000MHZ-ET/535-9995-1-ND/2001618>
- Ammar Mohammad, I., Ammar Mohammad, H., & Mohammad Abdel-Lateef, A.-T. (2014). Low Power, Small Area All Digital Phase Locked Loop (ADPLL). *Cairo University*.
- Best, R. E. (2003). *Phase Locked Loop. Desing, Simulation and Applications*. . Nueva York, USA.: Mc.Graw-Hill.
- Edway, S., & Manjunath, R. K. (2017). Design and Simulation of FPGA based All Digital Phase Locked Loop (ADPLL). *International Conference on Applied and Theoretical Computing and Commnication Technology*, 259-263.
- El-Rabbany, A. (2002). *Introduction to GPS. The Global Positioning System*. Noorwood, MA: Artech House.
- Gallardo Vázquez, S. (2015). Elementos de sistemas de telecomunicaciones. En S. Gallardo Vázquez, *Elementos de sistemas de telecomunicaciones* (págs. 48-65). Madrid, ESPAÑA: Paraninfo.
- Gaparini, L., Fontana, G., Macii, D., & Boni, A. (2007). A Digital Crcuit for Jitter Reduction of GPS-disciplined 1-pps Synchronization Signals. *AMUEM*.
- Gayathri, M. G. (2013). Design of All Digital Phase Locked Loop in VHDL. *International Journal of Engineering Research and Applications (IJERA)*, 3(4), 1074-1076.
- Ibrahim Afifiy, N., Abd El Mone'm, S. S., & Sayed Dahy, S. (2014). All Digital Phase Locked Loop (ADPLL). *Faculty of Engineering, Cairo Unniversity*.

- Instruments, N. (2011). Introducción a la Tecnología FPGA: Los Cinco Beneficios Principales. *Recuperado de: <http://www.ni.com/white-paper/6984/es/>*.
- Lata, K., & Kumar, M. (2013). ADPLL Design and Implementation on FPGA. *International Conference on Intelligent Systems and Signal Processing (ISSP)*.
- Machado Monteiro, C. A., de Souza, C. L., & Dalmas, M. (2016). Global Sample Synchronization Trigger. *IEEE International Symposium on Precision Clock Synchronization for Measurement, Control, and Communication (ISPCS)*, 1-4.
- McNeill, J. A., & Ricketts, D. (2009). The Designer's Guide to Jitter in Ring Oscillators. En J. A. McNeill, & D. Ricketts, *The Designer's Guide to Jitter in Ring Oscillators* (págs. 35-46). Boston, MA: Springer-Verlag US.
- Pérez Vega, C., Sáinz de la Maza, J. M., & Casanueva López, A. (2007). Sistemas de Telecomunicación. UNIVERSIDAD DE CANTABRIA.
- Tomasi, W. (2003). Sistemas de Comunicaciones Electrónicas. PEARSON EDUCACIÓN.
- Vydeki, N., & Srinivasa Rao, A. S. (2014). A Low Power VLSI Design of an All Digital Phase Locked Loop. *International Journal of Engineering Trends and Technology (IJETT)*.
- Walpole, R. E., Myers, R. H., Myers, S. L., & Ye, K. (2012). Probabilidad y estadística para ingeniería y ciencias . En R. E. Walpole, R. H. Myers, S. L. Myers, & K. Ye, *Probabilidad y estadística para ingeniería y ciencias* (págs. 172-185). México DF, México: Pearson Educación.

7. ANEXOS

7.1 Publicación de artículo

Pistas Educativas, No. 130, noviembre 2018, México, Tecnológico Nacional de México en Celaya

DISEÑO E IMPLEMENTACIÓN EN VHDL DE UN SISTEMA DE SINCRONIZACIÓN MEDIANTE FPGA y PPS PARA REFERENCIA DE 1 ms

DESIGN AND IMPLEMENTATION IN VHDL OF A SYNCHRONIZATION SYSTEM USING FPGA AND PPS FOR A 1 ms REFERENCE

María Amparo Sánchez Magaña

Universidad Autónoma de Querétaro
amparosanmag@gmail.com

Luis Morales Velázquez

Universidad Autónoma de Querétaro
lmorales@hspdigital.org

Resumen

Los lazos de seguimiento de fase (PLL) son ampliamente utilizados en materia de telecomunicaciones, redes inalámbricas y redes inteligentes como herramienta para mantener señales sincronizadas. En el presente trabajo se muestra el diseño propuesto de un lazo de seguimiento de fase completamente digital (ADPLL) utilizado como sintetizador de frecuencia. La solución desarrollada utiliza una referencia de temporización externa con alta estabilidad y a la salida obtiene una la sincronización de una señal con período de 1 ms.

Al modelo clásico del ADPLL se agregaron un par de módulos digitales con la finalidad de mantener las características de estabilidad del oscilador principal. Se comprobó además que mediante la implementación de un ADPLL se disminuye el error de fase acumulativo presente en todos los osciladores de cristal. Los resultados del modelo propuesto se muestran en la sección 3, así como también se presenta la simulación para verificar su funcionamiento y comprobar las ecuaciones propuestas en la sección 2.

Palabra(s) Clave: ADPLL, FPGA, GPS, sincronización.

Pistas Educativas Vol. 40 - ISSN: 2448-847X
Reserva de derechos al uso exclusivo No. 04-2016-120613261600-203
<http://itcelaya.edu.mx/ojs/index.php/pistas>

~1009~

Abstract

Phase Locked Loops (PLL) are widely used in telecommunication, wireless & wire line, and intelligent networks to keep signals synchronized. This paper shows the budget design for an All Digital Phase Locked Loop used as a frequency synthesizer. The develop solution uses an external high-stability timing reference that achieves the synchronization of a 1 ms period signal.

Two digital modules were added to the classic ADPLL model in order to maintain the stability characteristics of the main oscillator. It was also verified that through the implementation of an ADPLL it is possible to reduce the accumulative phase error present in all crystal oscillators. The results of the budget model are shown in section 3, as well as the simulation to verify the operation and verification of the equations developed in section 2.

Keywords: ADPLL, FPGA, GPS, synchronization.

1. Introducción

La sincronización en tiempo es un tema de gran relevancia hoy en día y su uso se ha extendido en las redes industriales en general. Normalmente estas redes están compuestas de una gran cantidad de nodos dispersos sobre un área con la finalidad de monitorear un fenómeno de interés, sin embargo, la recolección de los datos por cada nodo debe estar apropiadamente sincronizada con el objetivo de reducir variaciones que pueden ser perjudiciales en la adquisición de datos [Al-Kofahi, 2016]. La sincronización en tiempo normalmente se basa en marcas de tiempo que son proporcionadas por equipos que cuentan con una notable precisión y estabilidad que por ende son tomadas como referencia para la secuencia correcta de eventos en un proceso [Razo, 2016].

Gracias al avance tecnológico, los sistemas de posicionamiento global nos permiten sincronizar múltiples relojes en amplias áreas geográficas de manera exacta ya que está constituido por una constelación de 24 satélites con relojes atómicos, cuyos valores cuentan con una alta estabilidad y exactitud alcanzando fluctuaciones de período que rondan por debajo de las décimas de nanosegundos [Grobler, 2006].

Dentro de las aplicaciones relacionadas a la sincronización con tiempo global se pueden destacar aquellas en mediciones de voltaje en sistemas eléctricos de alta potencia [Yao, 2018], en redes de telecomunicación digital como herramienta para agregar flexibilidad en las mismas [Shan, 2014], por mencionar algunos ; donde generalmente para el ajuste en tiempo de una red se recurre a estándares y protocolos, establecidos por la IEEE (Institute of Electrical and Electronics Engineers), los cuales regulan el perfil de datos con la finalidad de cumplir con los requisitos de sincronización de tiempo [Amelot, 2010] tales como: IEEE 1588 Precision Time Protocol (PTP) o el Network Time Protocol (NTP). Sin embargo, en estos intervienen varios factores que no permiten que la sincronización se realice con precisión ya que a pesar del retardo de espera existen otros componentes de retardo como lo son el retardo de procesamiento y el retardo de transmisión [Hadzic, 2011]. Otra herramienta que logra proporcionar etiquetas de tiempo para las mediciones de eventos de manera uniforme son los lazos de seguimiento de fase (Phase Locked Loop, PLL) y que alcanza una alta resolución de etiquetas de tiempo, aunque requiere un tiempo de adquisición más largo para lograr una sincronización estable [Jahja, 2014].

Pese a que existen lazos de seguimiento analógicos, en el presente trabajo se describe el desarrollo de un lazo de seguimiento de fase completamente digital e implementado en FPGA cuyo detector de fase no depende de la superposición de la señal de referencia y la señal a sincronizar, sino que se pretende trabajar a partir de las comparaciones y no alineaciones de los flancos ascendentes de la señal de un pulso por segundo (Pulse Per Second, PPS) provista de un receptor GPS. Esta manera sincrónica de enlazar dos señales en frecuencia y fase permite continuar generando de forma correcta la señal a sincronizar aún y cuando el receptor se encuentre trabajando bajo condiciones no habituales, es decir, que la señal PPS se haya perdido.

El problema de la variación de frecuencia

En la actualidad los dispositivos electrónicos trabajan a una frecuencia nominal, la cual es proporcionada por un oscilador integrado, y que es especificada en la hoja

de datos del dispositivo. Sin embargo, en la realidad esta frecuencia no es constante, su estabilidad está ligada directamente a las condiciones de operación del dispositivo como la temperatura. Para determinar estas variaciones de frecuencia, se utiliza el término partes por millón (Parts-Per-Million, PPM), el cual se refiere a la diferencia entre la frecuencia nominal del oscilador y su frecuencia actual o real y es definido por el proveedor del oscilador [Cerdea, 2014].

Para la implementación del presente proyecto se utilizó una tarjeta de desarrollo DUA 4, la cual utiliza el oscilador 535-9995-1-ND de Abracon LLC con una frecuencia nominal de 48 MHz, con una variación de frecuencia especificada de +/- 100 ppm. Las frecuencias mínimas y máximas de operación se pueden calcular mediante las ecuaciones 1 y 2 respectivamente, con $f_{nominal} = 48MHz$ y $f_{nominal} = 48 MHz$.

$$f_{min} = \frac{f_{nominal}}{1 + \left(\frac{valor\ ppm}{1 \times 10^6}\right)} = 47,995,200.48\ Hz \quad (1)$$

$$f_{max} = f_{nominal} \left(1 + \frac{valor\ ppm}{1 \times 10^6}\right) = 48,004,800\ Hz \quad (2)$$

A simple vista, esta variación de frecuencia es muy pequeña y por lo tanto se supondría despreciable al no sobrepasar ni siquiera el 0.01% de error de la frecuencia nominal. Sin embargo, si se mantiene en operación continua esta deriva puede presentar errores en el rango de minutos en largos períodos de operación. Teniendo como ejemplo una señal de 1 Hz generada con un oscilador de 48 MHz trabajando únicamente a frecuencias mínimas, luego de treinta días de trabajo continuo la fluctuación total entre la señal generada y una señal de 1 Hz ideal serían $\Delta T = 259.14\ segundos$. O lo que es igual, la señal de 1 Hz generada por el oscilador estaría 4.31 minutos en atraso respecto a una señal ideal con la misma frecuencia.

Lazo de seguimiento de fase

El lazo de seguimiento de fase (Phase Locked Loop, PLL) es un sistema cerrado con retroalimentación capaz de mantener una señal de salida sincronizada tanto en frecuencia como en fase en relación con una señal de referencia [Roland, 2003].

Se trata de un circuito donde se pretende que el error de fase entre la señal de referencia y la señal de salida sea cero o se mantenga constante. El diagrama a bloques de un PLL básico consiste en tres bloques.

- Un detector de fase (PD – Phase Detector)
- Un filtro de lazo (LF – Loop Filter)
- Un oscilador de voltaje controlado (VCO – Voltage Controlled Oscillator)

Aunque un divisor de frecuencia se puede agregar cuando el PLL es utilizado como sintetizador de frecuencia, como se muestra en la figura 1.

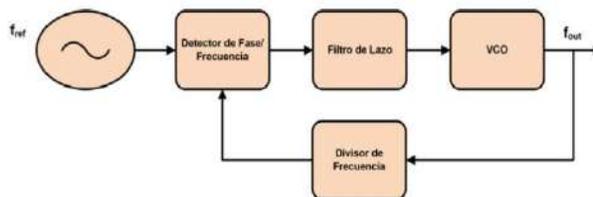


Figura 1 Diagrama general de PLL analógico.

Un detector de fase es un módulo responsable que comprara la fase de la señal de salida con la fase la señal de referencia. A la salida se obtiene una señal proporcional al error de fase θ_e , con un rango limitado al error máximo que puede haber entre ambas señales. La salida del módulo detector de fase puede modelarse mediante la ecuación 3.

$$u_d(t) = K_d \theta_e \quad (3)$$

donde k_d representa la ganancia del detector de fase y sus unidades están dadas en volts por radianes [Roland, 2003].

La señal u_d entra a la unidad filtro de lazo el cual se encarga de filtrar las frecuencias altas que no son deseadas, ya que el filtro de lazo se encarga de dejar pasar las frecuencias bajas y de suprimir las altas, por lo que se trata prácticamente de un filtro pasa-bajas. En la mayoría de los diseños de PLL se utiliza un filtro pasa-bajas de primer orden, algunas de las versiones más comúnmente encontradas son:

- Filtro pasivo de retardo de fase.

- Filtro activo de retardo de fase.
- Filtro activo PI (Proporcional-Integral).

Un oscilador de voltaje controlado (VCO) es un tipo de oscilador controlado muy utilizado en sistemas de comunicación en el que la frecuencia de oscilación es proporcional a un voltaje de entrada. Su función es la de convertir este voltaje de entrada de corriente directa (CD) a una señal sinusoidal en la salida, es decir que la frecuencia de la señal de salida cambia cuando el voltaje de entrada varía. En los PLL se encarga de oscilar a una frecuencia cercana a la frecuencia de la señal de entrada y varía su frecuencia de modo que se genere una señal que siga a la frecuencia de la señal de referencia. Los osciladores de voltaje controlado (VCO) son un tipo de oscilador controlado utilizados en variedad de aplicaciones como sistemas de comunicación y circuitos de temporización. Particularmente, los VCO's son empleados comúnmente en sistemas de control de lazo de seguimiento de fase. Se pueden considerar como circuitos que transforman una señal de voltaje de control de entrada en una señal de salida con una frecuencia deseada en el que la frecuencia de oscilación de la señal de salida es proporcional al voltaje de la señal de entrada [Roland, 2003]. Su función es la de convertir este voltaje de entrada de corriente directa (CD) a una señal sinusoidal en la salida, es decir que la frecuencia de la señal de salida cambia cuando el voltaje de entrada varía. En los PLL se encarga de oscilar a una frecuencia cercana a la frecuencia de la señal de entrada y varía su frecuencia de modo que se genere una señal que siga a la frecuencia de la señal de referencia. Los VCO's operan dentro de una banda finita de frecuencias, idealmente esta banda debe estar centrada alrededor de la frecuencia deseada, sin embargo, las variaciones inherentes del proceso limitan la precisión que se puede alcanzar al llevar a cabo el centrado de la banda de frecuencia cerca de la frecuencia que se desea [Mitchel, 2005].

2. Métodos

Se ha implementado la estructura mostrada en la figura 2, la cual consta principalmente de dos módulos:

- Módulo generador PPS.
- Módulo de sincronización.

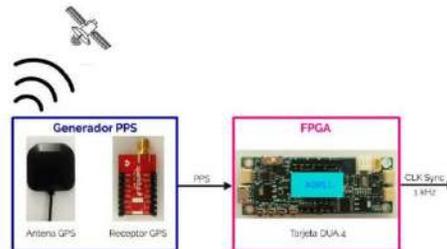


Figura 2 Esquema general de la metodología.

En el primero se encuentra la antena y el receptor GPS conectados entre ellos por medio de un conector SMA. Es importante mencionar que la antena GPS debe permanecer en todo momento en el exterior, con el fin de evitar fallas con el enlace satelital. El receptor GPS utilizado es el Venus634LPx el cual tiene a su salida una señal digital en estado lógico alto durante 4 ms a una frecuencia de 1 Hz. Esta señal es la referencia a la cual debe sincronizarse una señal generada por la tarjeta FPGA en el módulo de sincronización. Dicho módulo está constituido por un PLL completamente digital trabajando como sintetizador de frecuencia, por lo que se pretende generar una nueva señal a 1 kHz que mantenga las mismas características de estabilidad, exactitud y tolerancia del oscilador principal, logrando con esto que no se presente deriva en frecuencia ante períodos largos de operación.

La arquitectura digital propuesta del ADPLL se muestra en la figura 3, la cual consiste de un detector de flancos, un detector de fase-frecuencia, un oscilador digital controlado (DCO), un divisor de frecuencia, un contador y temporizador, además de una máquina de estados encargada de realizar la tarea de control de la estructura digital.

Como señal de entrada se tiene la señal PPS proveniente del GPS y como salida la señal CLK de 1 kHz sincronizada. La señal PPS ingresa al detector de flancos generando un pulso cada que este detecta un cambio de nivel lógico bajo a alto en

la señal de PPS. El pulso generado a partir de la señal PPS ingresa al detector de fase-frecuencia en donde se obtiene el desfase que esta señal presenta en relación con la señal generada por el divisor de frecuencia. La señal proveniente del divisor de frecuencia está constituida por un pulso de reloj a la misma frecuencia que la señal PPS generado a partir de la señal CLK. El desfase es representado por la variable K1 que sirve de valor final para el contador offset y las señales F1 y F2 indican si el desfase en la señal de 1 Hz está en adelante o en atraso. La entrada al contador proviene del temporizador offset el cual básicamente es la base de tiempo determinada por la diferencia de tiempo entre el periodo nominal y el mínimo o máximo esperado (K2). Una vez que el contador offset llega al valor establecido la compensación se considera concluida. La máquina de estados es la encargada de realizar la acción de control además del envío y recepción de señales de inicio y conclusión a algunos de los módulos involucrados.

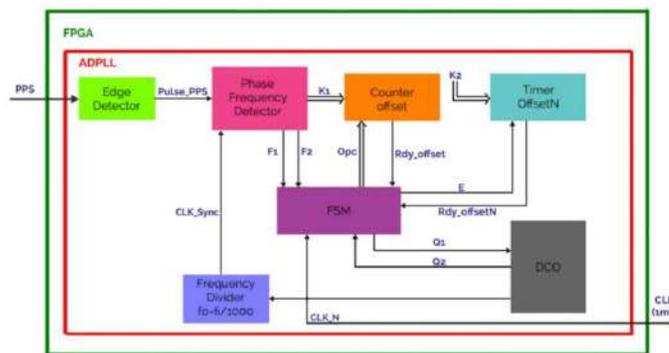


Figura 3 Diagrama a bloques ADPLL.

Detector de flancos

Como se mencionó anteriormente, los PLLs usaban el área de superposición entre las señales de entrada y la de retroalimentación para calcular el error de fase. No obstante, dentro de las estructuras digitales síncronas es necesario detectar la transición de un estado a otro. Para este caso en particular, un detector de flancos ascendentes en la señal PPS debe ser implementado.

Dicho detector de flancos está compuesto de dos Flip Flops tipo D, una compuerta lógica NOT y una compuerta lógica AND, tal y como se muestra en la figura 4. A la salida de este bloque se tendrá un pulso en alto cuando se detecte un flanco ascendente de la entrada PPS, en caso contrario la salida se mantendrá en bajo.

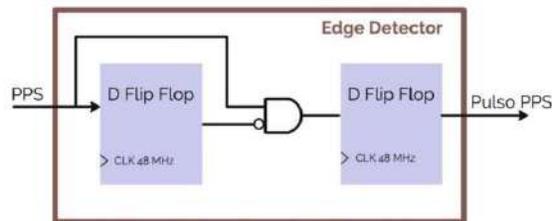


Figura 4 Diagrama a bloques, detector de flancos.

Debido a que ambos Flip Flops están conectados de manera serial, estos presentan un retardo en cada ciclo de la señal PPS proporcionada por el GPS, este retardo está dado en decenas de nanosegundos. El ángulo de fase entre la señal PPS y la señal pulso a la salida del módulo del detector de flancos está dado por la ecuación 4.

$$\varphi_{\text{detector flancos}} = \frac{2 * \pi}{2 * f_{osc}} = 2.62 \times 10^{-7} \text{ rad} \quad (4)$$

Este ángulo de fase entre ambas señales es constante y de ninguna manera se acumulará error entre ambas señales a través del tiempo.

Detector de fase- frecuencia

En el presente proyecto se diseñó un detector de fase que comparara los flancos ascendentes de las señales pulso PPS y la señal con frecuencia de 1 Hz a sincronizar (CLK Sync). A la salida de este módulo se encuentran dos banderas, F1 y F2, las cuales indican el estado entre las dos señales a enganchar:

- CLK Sync en atraso a la señal de referencia: Un estado lógico en alto se mantendrá en F1, mientras que F2 permanecerá en estado bajo.

- CLK Sync en adelante a la señal de referencia: Un estado lógico en alto se mantendrá en F₂, mientras que F₁ permanecerá en estado bajo.

Así mismo, a la salida se encuentra un número proporcional a la diferencia de tiempo entre los bordes ascendentes entre la señal de referencia entrante y la señal de retroalimentación; donde dicha cantidad se encuentra limitada a la tolerancia del oscilador de la tarjeta. Como se mencionó en el capítulo 1, el oscilador 535-9995-1-ND a utilizar indica una variación de frecuencia máxima de +/-100 ppm, la cual se pretende mantener en la señal CLK Sync. Aplicando la ecuación 1 y 2 se obtiene que, para una señal de 1 Hz, la variación de tiempo máxima entre la frecuencia nominal y la real que puede presentarse es de $\Delta T = 1 \times 10^{-4}$ segundos.

Para obtener el valor máximo a la salida del detector de fase frecuencia es necesario hacer un escalamiento e implementar un contador; donde el factor de escalamiento es la relación entre el desfase máximo deseado y la frecuencia del oscilador principal como se muestra en la ecuación 5.

$$K_1 = f_{oscilador} \Delta T = 4800 \text{ cuentas} \quad (5)$$

esta cuenta del contador es directamente proporcional a la magnitud del error de fase, tal y como se muestra en la ecuación 6.

$$\theta_e = \frac{2K\pi(f_{PFD})}{f_{oscilador} - f_{PFD}} \quad (6)$$

Por lo que el error de fase máximo que se puede compensar manteniendo la tolerancia del oscilador principal es de $\theta_{e(\max)} = 6.281876441 \times 10^{-4}$ rad.

DCO (Digital Controlled Oscillator)

Este módulo es de gran importancia en el lazo de seguimiento de fase y es la parte del sistema donde se realiza la compensación en tiempo de la señal a sincronizar.

Está compuesto por un temporizador operando a la frecuencia de resolución deseada y trabaja en conjunto con la máquina de estados principal. Posee dos

entradas: Q_1 y Q_2 , las cuales modifican directamente la cuenta del temporizador del DCO.

En la estructura digital, un temporizador se describe como un contador de ciclos del oscilador y un comparador. El contador se encuentra activo en todo momento y cuando la cuenta actual logre alcanzar la cantidad necesaria para obtener una señal de 1 ms, un pulso en estado lógico alto (CLK_N) se mantendrá a la salida del DCO. La cuenta del temporizador puede ser directamente modificada mediante las señales de entrada Q_1 y Q_2 . La entrada Q_1 representa un atraso de la señal CLK Sync con referencia a la señal pulso PPS por lo que la base de tiempo del temporizador deberá disminuirse. Esto se logra aumentando la tasa de cambio a dos mientras Q_1 esté en estado lógico alto.

En el caso contrario que se presente un adelanto de la señal CLK Sync con referencia a la señal pulso PPS la base de tiempo del temporizador deberá aumentarse, manteniendo la cuenta actual del temporizador mientras Q_2 esté en estado lógico alto. Este principio de funcionamiento puede verse en la tabla 1. Donde cuenta(n) representa la cuenta actual en el contador del temporizador del DCO.

Tabla 1 Tabla de verdad contador del DCO.

Entradas		Cuenta (n+1)
Q_1	Q_2	
0	0	Cuenta(n)+1
0	1	Cuenta(n)
1	0	Cuenta(n)+2

Contador offset y temporizador offsetN

Como se explicó en la descripción del detector de fase frecuencia, existe factor máximo para el conteo del desfase entre la señal de entrada y de retroalimentación de 1 Hz. No obstante, este factor no es el mismo para una señal con una frecuencia mil veces mayor. Se espera que la señal de 1 ms a la salida del módulo DCO cuente con las mismas características de estabilidad que el oscilador principal, es por ello por lo que un módulo extra (temporizador offsetN) es agregado al diseño.

Calculando la variación de tiempo máxima en una señal de 1 kHz con tolerancia de +-100 ppm, $\Delta T (\max) = 1 \times 10^{-7}$ segundos.

El cual corresponde a un factor de: $K_2 (\max) = 4.8 \approx 5$

Debido a que el factor no es un número fraccionario, se redondea hacia el entero próximo. Si bien esto nos permite lograr la compensación máxima necesaria, la tolerancia para una señal con frecuencia de 1 kHz aumentará en relación al aumento del factor en el temporizador offsetN, como se muestra en la ecuación 7.

$$PPM = \frac{1 \times 10^6 \text{ factor}}{f_{oscilador} T_{nominal}} = 104.1666667 \quad (7)$$

El módulo temporizador offsetN es un temporizador que rige la compensación en la señal de 1 kHz, mientras que el contador offset cuenta la compensación máxima en la señal de 1 Hz. La tabla de verdad del contador se muestra en la tabla 2.

Tabla 2 Tabla de verdad contador del contador offset.

Opc	Q(n+1)
00	Q(n)
01	Q(n)
10	Q(n)+1
11	'0'

Divisor de frecuencia

Por último, pero no menos importante se encuentra un divisor de frecuencia, el cual, como su nombre lo indica divide la frecuencia de la señal obtenida por el DCO. Recordemos que a la salida del DCO se tiene un pulso que tiene una frecuencia equivalente a la resolución que se requiere (1 ms) y que por razones obvias no puede compararse en frecuencia con la señal pulso PPS proporcionada por el GPS. Es por esto que un divisor de frecuencia es necesario antes de ser retroalimentado al detector fase frecuencia.

La señal a la salida del DCO debe ser escalada por un factor N, demostrado en la ecuación 8, el cual controlará la frecuencia a la salida del módulo divisor de frecuencia.

$$f_{sync} = N f_{CLKN} \quad (8)$$

Para este caso en particular, una frecuencia de 1 kHz debe ser escalada a la misma frecuencia de 1 Hz del PPS por lo que $N=1000$.

3. Resultados

En una primera prueba, se pretendió conocer el comportamiento del oscilador 535-9995-1-ND de Abracon LLC. Se generó una señal de 1 Hz sin utilizar ningún método de sincronización. Dentro de las primeras muestras ambas señales presentaron un error de fase mínimo, casi despreciable. Luego de veinte minutos de operación constante, se pudo observar un desfase mayor entre las señales, alcanzando los 1.288 ms tal y como se muestra en la figura 5. El canal C1 muestra la señal generada por el FPGA mientras que la señal C2 es la señal provista por el receptor GPS. Si bien se puede concluir que el oscilador de la tarjeta se encuentra trabajando dentro de la tolerancia especificada, la deriva en el tiempo que presenta se observa como un error de fase acumulativo a lo largo del tiempo.



Figura 5 Medición de desfase entre señales sin ADPLL después 20 minutos de operación ($\Delta T=1.2$ ms).

La prueba anterior se repitió una vez más, utilizando esta vez el módulo de sincronización diseñado. Dentro de este módulo la señal es retroalimentada con el fin de disminuir el error de fase acumulativo. Los resultados obtenidos se muestran en la figura 6. Durante los veinte minutos de operación ambas señales presentaron un desfase mínimo, no acumulativo y menor a $1 \mu s$. Por lo tanto, el ADPLL

diseñado logra mantener enganchadas ambas señales conservando el ángulo entre ellas por debajo de la tolerancia máxima especificada.

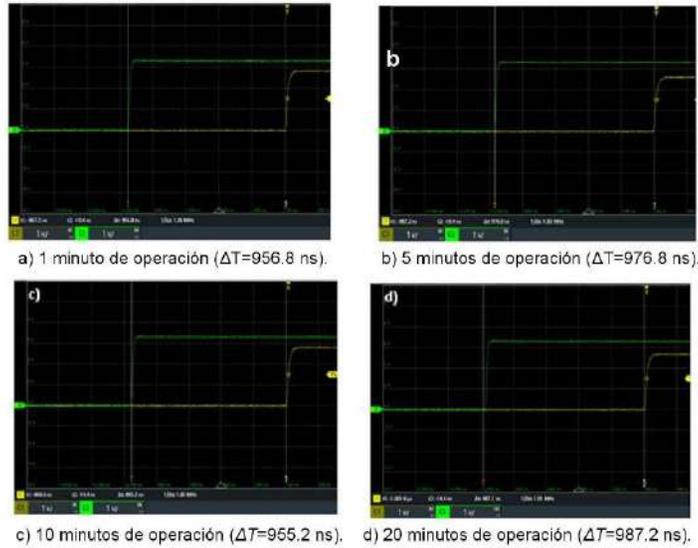


Figura 6 Medición de desfase entre señales después de

Se realizó, además, la simulación del sistema con la finalidad de observar el comportamiento en la señal de 1 kHz. Se simuló un escenario en el cual ambas señales presentarían una diferencia en su período de $\Delta T=1.6146 \times 10^{-5}$ segundos (figura 7).

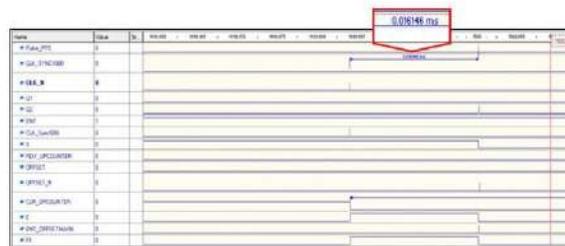


Figura 7 Desfase entre CLK Sync y Pulso PPS corresponde a $\Delta T=1.6146 \times 10^{-5}$ segundos.

redondeo que presentó la cuenta del temporizador que rige la compensación para 1 kHz. La tolerancia tuvo que ser ajustada a un nuevo valor de ± 104.16 PPM. Esto podría corregirse utilizando un oscilador con una frecuencia múltiplo de $\frac{1}{\Delta T}$.

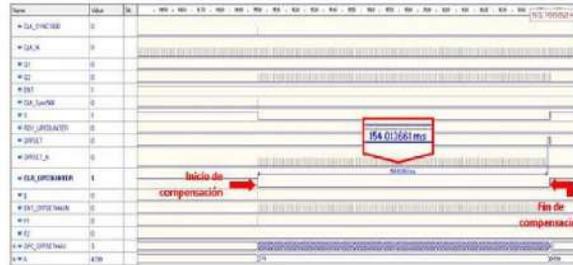


Figura 9 Duración de la compensación total.

Tal y como se mostró en los resultados, la diferencia de fase entre la señal de referencia y la de retroalimentación representa el porcentaje de ciclos de la señal de 1 kHz que serán compensados la cantidad de $\Delta T_{\max(1 \text{ kHz})}$. A pesar de que esta compensación mantiene a dicha señal dentro de los límites de tolerancia y estabilidad esperados, nuestro diseño podría mejorarse dividiendo la compensación total entre cada uno de los ciclos de CLK N.

5. Conclusiones

Se diseñó un lazo de seguimiento de fase completamente digital que permite sincronizar una señal de 1 Hz generada por el FPGA con una señal de poca fluctuación provista por un módulo GPS.

Gracias al diseño propuesto, este módulo de sincronización trabaja a partir de los flancos ascendentes de las señales a sincronizar; por lo que, si el GPS perdiera la conexión satelital, el ADPLL no realiza compensaciones innecesarias que podrían llegar a afectar la estabilidad del reloj generado.

La señal de 1 Hz obtenida muestra poca derivación en el tiempo y mantiene un bajo error de fase incluso en largos períodos de tiempo.

Además, en la implementación mostró una tolerancia considerablemente menor a la esperada. Se generó una señal con una frecuencia 1000 veces mayor a la señal de referencia. Sin embargo, debido al redondeo en el factor del temporizador para generar dicha señal la tolerancia en la señal de 1 KHz aumentó un 4.16%. Debido a que el factor en el temporizador de 1 Hz es una cantidad fija, la compensación en esta señal no se realiza de manera uniforme dentro de los N ciclos, sino que el error entre la señal generada y la señal de referencia de 1 Hz representa el porcentaje de ciclos a los cuales se les asignará el 100% de la compensación.

6. Bibliografía y Referencias

- [1] Al-Kofahi, O., Evaluating time synchronization using application-layer time-stamping, 2016 IEEE Wireless Communications and Networking Conference, Doha, pp. 1-6, 2016.
- [2] Amelot, J., Fletcher, J., Anand, D., Vasseur, C., Li-Babound, Y. S. and Moyne, J., An IEEE 1588-time synchronization testbench for assesing power distribution requirements, 2010 IEEE International Symposium on Precision Clock Synchronization for Measuments, Control and Communication, Portsmouth, NH, pp. 13-18, 2010.
- [3] Calderón Zavala, G., Mina Antonio, J. D., López Núñez, A. R. and Martínez Flores, L. R., Modelado de un generador de SAGS basado en un convertidor back to back, 2015 SENIE.
- [4] Grobler, M. and Naidoo, R., Determining Transmission Line Parameters from GPS Time-Stamped Data, IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics, Paris, pp. 3580-3584, 2006.
- [5] Hadvzic, I., Morgan, D. R., and Sayeed, Z., A Synchronization Algorithm for Packet MANs, in IEEE Transactions on Communications, vol. 59, no. 4, pp. 1142-1153, April 2011.
- [6] Razo-Hernandez, J. R., Valtierra-Rodriguez, M., Lieberman, D. G., Amezquita-Sanchez J. P., Morales-Hernandez, L. A. and Dominguez-Gonzalez, A. IRIG-B decoder based on FPGA for synchronization in PMUs

- by considering different formats, 2016 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC), Ixtapa, pp.1-6, 2016.
- [7] Jahja, R. H., Dahal, S., Suk-seung, H., Goo-Rak, K., Pyun, J. Y. and Shin, S. Improving IEEE 1588v2 time synchronization performance with phase locked loop, 2014 48th Asilomar Conference of Signals, Systems and Computers, Pacific Grove, CA, pp. 252-257, 2014.
- [8] M. Cerda, Ramón. Understanding Quartz Crystals and Oscillators, Editorial Artech House, 2014.
- [9] P. Mitchel, Eric, R. Gehring, Mark. Voltage Controlled Oscillator, United States Patent, 2005.
- [10] Roland E. Best, Phase Locked Loops. Design, Simulation and Applications, Quinta edición, McGraw-Hill, 2003.
- [11] Shan, Q., Jun, Y., Le Floch, J. M., Fan, Y., Ivanov, E. N. and Tobar, M. E., Simulating GPS radio signal to synchronize network – a new technique for redundant timing, IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, vol. 61, no. 7, pp. 1075-1085, July 2014.
- [12] Yao, W., Lu, H., Till, M. J., Gao, W. and Liu, Y., Synchronized Wireless Measurement of High-Voltage Power System Frequency Using Mobile Embedded Systems, in IEEE Transactions on Industrial Electronics, vol. 65, no. 3, pp. 2775-2784, March 2018.

7.2 Constancia de presentación de artículo

Universidad
Autónoma
Metropolitana
Azcapotzalco



SENTE 2018



La Universidad Autónoma Metropolitana-Azcapotzalco a través de la
División de Ciencias Básicas e Ingeniería y la Universidad Panamericana Campus Bonaterra

Otorgan la presente

CONSTANCIA A:

María Amparo Sánchez Magaña

Por haber participado como ponente del trabajo titulado:

**Diseño e implementación en VHDL de un sistema de sincronización mediante FPGA y PPS para
referencia de 1 ms**

Presentado en la XIV Semana Nacional de Ingeniería Electrónica

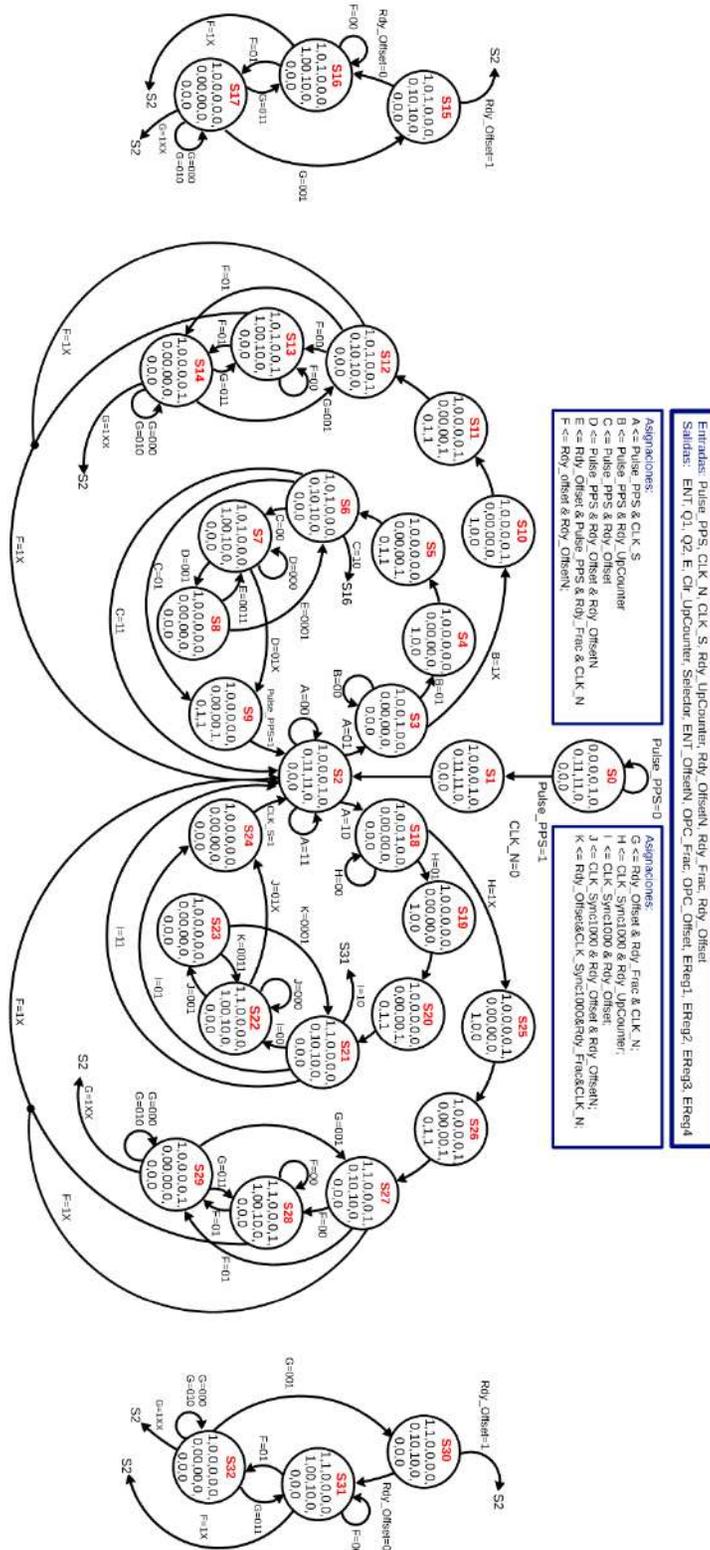
Aguascalientes, Ags. a 5 de octubre de 2018.

Dra. María de Lourdes Delgado Núñez
Directora de la División de CBI
Universidad Autónoma Metropolitana-Azcapotzalco

Dr. Miguel Magos Rivera
Presidente del Comité Organizador
Sente 2018

Dr. Ramón Velázquez Guerrero
Vicepresidente de Investigación
Universidad Panamericana - Aguascalientes

7.3 Máquina de estados, detector de fase frecuencia



7.4 Módulo detector de fase frecuencia

```
library ieee;
use ieee.std_logic_1164.all;
```

```
entity Phase_Frequency_Detector is
  port(
    CLK                : in std_logic;
    RST                : in std_logic;
    PULSE_PPS          : in std_logic;
    CLK_N              : in std_logic;
    CLK_Sync1000       : in std_logic;
    RDY_UPCOUNTER      : in std_logic;
    RDY_OFFSETN        : in std_logic;
    RDY_FRAC           : in std_logic;
    RDY_OFFSET         : in std_logic;
    ENT                : out std_logic;
    Q1                 : out std_logic;
    Q2                 : out std_logic;
    E1                 : out std_logic;
    CLR_UPCOUNTER      : out std_logic;
    SELECTOR           : out std_logic;
    ENT_OFFSETN        : out std_logic;
    OPC_FRAC           : out std_logic_vector (1 downto 0);
    OPC_OFFSET         : out std_logic_vector (1 downto 0);
    EREG1              : out std_logic;
    EREG2              : out std_logic;
    EREG3              : out std_logic;
    EREG4              : out std_logic
  );
end Phase_Frequency_Detector;
```

```
architecture behavior of Phase_Frequency_Detector is
  type FSM_var is (S0, S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11, S12, S13, S14, S15, S16, S17,
  S18, S19, S20, S21, S22, S23, S24, S25, S26, S27, S28, S29, S30, S31, S32);
  Signal Qp, Qn: FSM_var;
  Signal A, B, C, F, H, I: std_logic_vector (1 downto 0);
  Signal D, G, J: std_logic_vector (2 downto 0);
  Signal E, K: std_logic_vector(3 downto 0);
  Signal AUX: std_logic_vector (14 downto 0);
begin
```

```
  --OUTPUT SIGNALS
  ENT                <=  AUX(14);
  Q1                 <=  AUX(13);
  Q2                 <=  AUX(12);
  E1                 <=  AUX(11);
  CLR_UPCOUNTER      <=  AUX(10);
  SELECTOR           <=  AUX( 9);
  ENT_OFFSETN        <=  AUX( 8);
  OPC_FRAC           <=  AUX( 7 downto 6);
  OPC_OFFSET         <=  AUX( 5 downto 4);
  EREG1              <=  AUX( 3);
  EREG2              <=  AUX( 2);
  EREG3              <=  AUX( 1);
```

```
EREG4 <= AUX( 0);
```

```
--AUXILIARY SIGNALS
```

```
A <= Pulse_PPS&CLK_Sync1000;  
B <= Pulse_PPS&Rdy_UpCounter;  
C <= Pulse_PPS&Rdy_Offset;  
D <= Pulse_PPS&Rdy_Offset&Rdy_OffsetN;  
E <= Rdy_Offset&Pulse_PPS&Rdy_Frac&CLK_N;  
F <= Rdy_offset&Rdy_OffsetN;  
G <= Rdy_Offset&Rdy_Frac&CLK_N;  
H <= CLK_Sync1000&Rdy_UpCounter;  
I <= CLK_Sync1000&Rdy_Offset;  
J <= CLK_Sync1000&Rdy_Offset&Rdy_OffsetN;  
K <= Rdy_Offset&CLK_Sync1000&Rdy_Frac&CLK_N;
```

```
Qp<=S0 when RST='1' else Qn when rising_edge(CLK);
```

```
K) process(Qp, PULSE_PPS, CLK_N, Rdy_Offset, CLK_Sync1000, A, B, C, D, E, F, G, H, I, J,  
begin
```

```
    case Qp is  
        when S0 =>  
            if PULSE_PPS='0' then  
                Qn<= Qp;  
            else  
                Qn<= S1;  
            end if;  
            AUX<="000010011110000";  
  
        when S1 =>  
            if CLK_N='0' then  
                Qn<= Qp;  
            else  
                Qn<= S2;  
            end if;  
            AUX<="100010011110000";  
  
        when S3 =>  
            case B is  
                when "00" =>  
                    Qn<= Qp;  
                when "01" =>  
                    Qn<= S4;  
                when "10" =>  
                    Qn<= S10;  
                when others =>  
                    Qn<= S10;  
            end case;  
            AUX<="100100000000000";  
  
        when S4 =>  
            Qn <= S5;  
            AUX<="100000000000100";  
  
        when S5 =>
```

```
Qn <= S6;
AUX<="100000000001011";
```

```
when S6          =>
case C is
  when "00"      =>
    Qn<= S7;
  when "01"      =>
    Qn<= S9;
  when "10"      =>
    Qn<= S16;
  when others =>
    Qn<= S2;
```

```
end case;
AUX<="101000010100000";
```

```
when S7          =>
case D is
  when "000"     =>
    Qn<= Qp;
  when "001"     =>
    Qn<= S8;
  when "010"     =>
    Qn<= S9;
  when "011"     =>
    Qn<= S9;
  when "100"     =>
    Qn<= S16;
  when "101"     =>
    Qn<= S17;
  when "110"     =>
    Qn<= S2;
  when others =>
    Qn<= S2;
```

```
end case;
AUX<="101000100100000";
```

```
when S8          =>
case E is
  when "0000"    =>
    Qn<= Qp;
  when "0001"    =>
    Qn<= S6;
  when "0010"    =>
    Qn<= Qp;
  when "0011"    =>
    Qn<= S7;
  when "0100"    =>
    Qn<= S17;
  when "0101"    =>
    Qn<= S15;
  when "0110"    =>
    Qn<= S17;
  when "0111"    =>
    Qn<= S16;
```

```

        when others =>
            Qn<= S2;
        end case;
AUX<="1000000000000000";

    when S9          =>
        if PULSE_PPS='0' then
            Qn<= Qp;
        else
            Qn<= S2;
        end if;
        AUX<="1000000000000000";

    when S10         =>
        Qn <=S11;
        AUX<="100001000000100";

    when S11         =>
        Qn <=S12;
        AUX<="100001000001011";

    when S12         =>
        case F is
            when "00"    =>
                Qn<= S13;
            when "01"    =>
                Qn<= S14;
            when "10"    =>
                Qn<= S2;
            when others =>
                Qn<= S2;
        end case;
        AUX<="101001010100000";

    when S13         =>
        case F is
            when "00"    =>
                Qn<= Qp;
            when "01"    =>
                Qn<= S14;
            when "10"    =>
                Qn<= S2;
            when others =>
                Qn<= S2;
        end case;
        AUX<="101001100100000";

    when S14         =>
        case G is
            when "000"   =>
                Qn<= Qp;
            when "001"   =>
                Qn<= S12;
            when "010"   =>
                Qn<= Qp;

```

```

        when "011" =>
            Qn<= S13;
        when others =>
            Qn<=S2;
    end case;
    AUX<="1000010000000000";

```

```

    when S15 =>
        if Rdy_Offset='0' then
            Qn<= S16;
        else
            Qn<= S2;
        end if;
    AUX<="101000010100000";

```

```

    when S16 =>
        case F is
            when "00" =>
                Qn<= Qp;
            when "01" =>
                Qn<= S17;
            when "10" =>
                Qn<= S2;
            when others =>
                Qn<= S2;
        end case;
    AUX<="101000100100000";

```

```

    when S17 =>
        case G is
            when "000" =>
                Qn<= Qp;
            when "001" =>
                Qn<= S15;
            when "010" =>
                Qn<= Qp;
            when "011" =>
                Qn<= S16;
            when others =>
                Qn<= S2;
        end case;
    AUX<="1000000000000000";

```

```

    when S18 =>
        case H is
            when "00" =>
                Qn<= Qp;
            when "01" =>
                Qn<= S19;
            when "10" =>
                Qn<= S25;
            when others =>
                Qn<= S25;
        end case;
    AUX<="1001000000000000";

```

```
when S19    =>
Qn <=S20;
AUX<="100000000000100";
```

```
when S20    =>
Qn <=S21;
AUX<="100000000001011";
```

```
when S21    =>
case I is
  when "00"    =>
    Qn<= S22;
  when "01"    =>
    Qn<= S24;
  when "10"    =>
    Qn<= S31;
  when others =>
    Qn<= S2;
end case;
AUX<="110000010100000";
```

```
when S22    =>
case J is
  when "000"   =>
    Qn<= Qp;
  when "001"   =>
    Qn<= S23;
  when "010"   =>
    Qn<= S24;
  when "011"   =>
    Qn<= S24;
  when "100"   =>
    Qn<= S31;
  when "101"   =>
    Qn<= S32;
  when "110"   =>
    Qn<= S2;
  when others =>
    Qn<= S2;
end case;
AUX<="110000100100000";
```

```
when S23    =>
case K is
  when "0000"  =>
    Qn<= Qp;
  when "0001"  =>
    Qn<= S21;
  when "0010"  =>
    Qn<= Qp;
  when "0011"  =>
    Qn<= S22;
  when "0100"  =>
    Qn<= S32;
```

```

        when "0101" =>
            Qn<= S30;
        when "0110" =>
            Qn<= S32;
        when "0111" =>
            Qn<= S31;
        when others =>
            Qn<= S2;
    end case;
    AUX<="1000000000000000";

```

```

    when S24 =>
        if CLK_Sync1000='0' then
            Qn<= Qp;
        else
            Qn<= S2;
        end if;
    AUX<="1000000000000000";

```

```

    when S25 =>
        Qn <=S26;
    AUX<="100001000000100";

```

```

    when S26 =>
        Qn <=S27;
    AUX<="100001000001011";

```

```

    when S27 =>
        case F is
            when "00" =>
                Qn<= S28;
            when "01" =>
                Qn<= S29;
            when "10" =>
                Qn<= S2;
            when others =>
                Qn<= S2;
        end case;
    AUX<="110001010100000";

```

```

    when S28 =>
        case F is
            when "00" =>
                Qn<= Qp;
            when "01" =>
                Qn<= S29;
            when "10" =>
                Qn<= S2;
            when others =>
                Qn<= S2;
        end case;
    AUX<="110001100100000";

```

```

    when S29 =>
        case G is

```

```

        when "000"    =>
            Qn<= Qp;
        when "001"    =>
            Qn<= S27;
        when "010"    =>
            Qn<= Qp;
        when "011" =>
            Qn<= S28;
        when others =>
            Qn<= S2;
    end case;
    AUX<="1000010000000000";

    when S30    =>
    if Rdy_Offset='1' then
        Qn<= S2;
    else
        Qn<= S31;
    end if;
    AUX<="1100000101000000";

    when S31    =>
    case F is
        when "00"    =>
            Qn<= Qp;
        when "01"    =>
            Qn<= S32;
        when "10"    =>
            Qn<= S2;
        when others =>
            Qn<= S2;
    end case;
    AUX<="1100001001000000";

    when S32    =>
    case G is
        when "000"    =>
            Qn<= Qp;
        when "001"    =>
            Qn<= S30;
        when "010"    =>
            Qn<= Qp;
        when "011" =>
            Qn<= S31;
        when others =>
            Qn<= S2;
    end case;
    AUX<="1000000000000000";

    when others => --S2
    case A is
        when "00"    =>
            Qn<= Qp;
        when "01"    =>
            Qn<= s3;
    end case;

```

```
        when "10"      =>
            Qn<= s18;
        when others =>
            Qn<= Qp;
    end case;
    AUX<="100010011110000";
end case;
end process;
end behavior;
```