

2012

Evaluación de los parámetros de desempeño de un convertidor
analógico – digital Δ - Σ utilizando técnicas de procesamiento digital
de señales
Ing. Marco Antonio Rodríguez Guerrero.



Universidad Autónoma de Querétaro
Facultad de Ingeniería

Evaluación de los parámetros de desempeño de un
convertidor analógico – digital Δ - Σ utilizando técnicas de
procesamiento digital de señales

Tesis

Que como parte de los requisitos para obtener el grado de

Maestro en Ciencias

en Instrumentación y Control Automático

Presenta

Ing. Marco Antonio Rodríguez Guerrero.

Santiago de Querétaro, Querétaro, México.

Mayo 2012.



Universidad Autónoma de Querétaro.
Facultad de Ingeniería.
Maestría en Ciencias
en Instrumentación y Control Automático.

**Evaluación de los parámetros de desempeño de convertidor analógico – digital $\Delta\Sigma$
utilizando técnicas de procesamiento digital de señales.**

Tesis.

Que como parte de los requisitos para obtener el grado de
Maestro en Ciencias en Instrumentación y Control Automático.

Presenta:

Ing. Marco Antonio Rodríguez Guerrero.

Dirigido por:

Dr. René de Jesús Romero Troncoso.

Dr. René David Carranza López Padilla.

SINODALES

Dr. René de Jesús Romero Troncoso

Presidente.

Dr. Roque Alfredo Osornio Ríos

Secretario

Dr. Manuel Toledano Ayala

Vocal

Dr. Luis Morales Velázquez

Suplente

Dr. René David Carranza López Padilla

Suplente

Dr. Aurelio Dominguez González.

Director de la Facultad de Ingeniería

Firma

Firma

Firma

Firma

Firma

Dr. Irineo Torres Pacheco

Director de Investigación y
Posgrado.

Centro Universitario

Santiago de Querétaro, Querétaro. Mayo de 2012.

México.

RESUMEN

En la presente investigación se realiza la caracterización metrológica de un convertidor de datos analógico – digital de tecnología delta – sigma (Δ - Σ) de alta resolución aplicado a mediciones eléctricas de alta exactitud. Utilizando este convertidor Δ - Σ se desarrolló un sistema de medición de tensión eléctrica continua y tensión eléctrica alterna, el cual fue evaluado en sus principales parámetros de desempeño, como lo son linealidad en todo el intervalo dinámico, relación señal a ruido efectiva, resolución máxima efectiva, estabilidad etc. Durante la caracterización metrológica del convertidor Δ - Σ se aplican algoritmos que permiten comparar el desempeño del convertidor Δ - Σ respecto de otros sistemas de medición de referencia metrológica de mayor jerarquía. Los parámetros más importantes del convertidor analógico – digital se estudiaron aplicando directamente el convertidor analógico – digital como medidor de tensión eléctrica continua y tensión eléctrica alterna; en ambos casos, el medidor muestra alta resolución (mayor a 21 bits), alta relación señal a ruido (menor a -120 dB) y alta estabilidad (mejor que 1 μ V/V en 24 horas). Para desarrollar dicha comparación fue necesario aplicar técnicas de procesamiento digital de señales, como lo son la propuesta de filtros digitales implementados sobre lógica reconfigurable FPGA. Se presentan los resultados comparativos y se realiza una propuesta de uso del convertidor de datos.

(Palabras clave: Convertidor Analógico-Digital, ADC, Delta-Sigma, DSP, HSP, FPGA, Metrología Eléctrica, Mediciones de Alta exactitud.)

SUMMARY

Present work develops a metrology characterization about an analog to digital data converter based on sigma – delta technology in high resolution, all this applied to high accuracy electromagnetic measurements. Using a Δ - Σ ADC it was developed an AC & DC voltage measurement system, which was evaluated based on its performance parameters such as dynamic range linearity, signal-to-noise ratio, effective resolution, short term stability, etc.

During the metrological characterization, processing signal algorithms were applied in order to compare the ADC's performance against better metrological systems. Main performance parameter were studied using the ADC directly as AC & DC voltage meter; for both cases, the developed meter shows high resolution (better than 21-bit), high signal-to-noise ratio (better than -120 dB) and high stability (better than 1 μ V/V in 24 hrs). In order to develop this comparison it was necessary suggest digital signal processing techniques, like filtering algorithms. A result comparative and a possible uses of the ADC in electric metrology are given.

(Key words: Data converters, ADC, DSP, Sigma-Delta, Electrical Metrology, High Accuracy Measurements, Hardware Signal Processing, FPGA)

AGRADECIMIENTOS

El autor desea hacer patente su agradecimiento al Centro Nacional de Metrología y a su Sistema de Desarrollo Profesional (SIDEPRO) por el apoyo para la realización de este trabajo de investigación en el área de metrología eléctrica, así como a la Universidad Autónoma de Querétaro por las facilidades otorgadas para el desarrollo del presente trabajo.

Al Dr. René David Carranza López Padilla por el invaluable apoyo y la dirección para que este trabajo llegara a buen término.

Al Dr. René de Jesús Romero Troncoso, por sus consejos, por su paciencia y por sus atinados comentarios.

Al personal académico del posgrado de la Facultad de Ingeniería de la Universidad Autónoma de Querétaro, por su contribución en este trabajo.

DEDICATORIAS

Dedico este trabajo a...

Mariela, por tu amor y paciencia, sin tí no lo hubiera logrado.

A mis padres y hermanos, por la fe depositada en mí.

A mi familia y amigos más cercanos, por estar siempre pendientes.

A Sara, por el empujón inicial.

Al grupo de trabajo, Sergio, Daniel y Adrián, por su constante animo, por los momentos vividos en el laboratorio y por sus consejos a través de estos años.

Esto es para ustedes, Gracias.

Ad astra per aspera...

ÍNDICE

<i>INDICE DE TABLAS</i>	<i>ix</i>
<i>INDICE DE FIGURAS</i>	<i>x</i>
1. INTRODUCCIÓN.	1
1.1 <i>Objetivo.</i>	4
1.2 <i>Hipótesis.</i>	4
1.3 <i>Reto del proyecto.</i>	5
1.4 <i>Justificación.</i>	6
2. REVISIÓN BIBLIOGRÁFICA.	7
2.1 <i>Convertidores analógico-digitales con tecnología Δ-Σ.</i>	7
2.2 <i>Sistemas de muestreo en aplicaciones de metrología eléctrica.</i>	13
2.3 <i>Estimación de valores RMS a partir de muestreo no síncrono.</i>	16
3. METODOLOGÍA.	18
3.1 <i>Determinación del valor RMS de una señal periódica a partir de un método de ventana.</i>	22
3.2 <i>Determinación del valor RMS de una señal periódica a partir del algoritmo Swerlein.</i>	26
3.3 <i>Normas internacionales para términos, definiciones y métodos de prueba de convertidores analógico-digital.</i>	28
3.4 <i>Diseño del sistema de muestreo digital utilizando un ADC Δ-Σ.</i>	31
3.5 <i>Fuente de Referencia de corriente directa de ± 2.5 V.</i>	32
3.6 <i>Convertidor analógico-digital Δ-Σ (Texas Instruments ADS1281).</i>	35
3.7 <i>Aislamiento digital. Protección contra ruido radiado y conducido hacia el ADC Δ-Σ.</i>	37

3.8	Reloj maestro de 4.096 MHz.....	38
3.9	Amplificador de entrada completamente diferencial en el ADC Δ - Σ	39
3.10	Control y procesamiento de señales para mediciones de niveles de tensión eléctrica continua y tensión eléctrica alterna basado en un FPGA.	43
3.11	Diseño de experimentos para la evaluación de los parámetros de desempeño del convertidor analógico – digital Δ - Σ	51
4.	<i>RESULTADOS</i>	52
4.1	Evaluación del hardware diseñado para el sistema de muestreo digital basado en un ADC Δ - Σ ADS1281.	52
4.2	Resultados de la caracterización de la fuente de referencia a partir de un Zener de 10 V Fluke 732B.	56
4.3	Resultados de la caracterización de ruido del convertidor analógico-digital ADS1281.....	57
4.4	Resultados de la caracterización de los aisladores digitales.	58
4.5	Resultados de la caracterización del amplificador de instrumentación completamente diferencial.	60
4.6	Resultados de la caracterización del algoritmo para la determinación del valor rms de una señal periódica de baja frecuencia.	64
4.7	Resultados de la caracterización del sistema de muestreo digital comparado contra sistemas de referencia para medición de tensión eléctrica alterna y tensión eléctrica continua con alta exactitud.....	66
5.	<i>DISCUSIÓN DE RESULTADOS</i>	73
6.	<i>CONCLUSIONES</i>	75
7.	<i>BIBLIOGRAFÍA</i>	77
	<i>ANEXO I. Código VHDL para las principales etapas de procesamiento.</i>	80

INDICE DE TABLAS

Tabla 3.1	Detalle de los instrumentos comerciales utilizados durante el presente trabajo de investigación.	19
Tabla 4.1	Resultados de la caracterización de ruido del amplificador de instrumentación completamente diferencial utilizando el algoritmo Swerlein.	61
Tabla 4.2	Diferencias máximas por linealidad en amplificador completamente diferencial.	63
Tabla 4.3	Resultados de la caracterización del algoritmo para la determinación de valores RMS utilizando ventana Hanning.	65
Tabla 4.4	Resultados de la caracterización de la resolución efectiva del sistema de muestreo basado en un convertidor analógico-digital Δ - Σ .	67
Tabla 4.5	Resultados experimentales de la comparación de instrumentación comercial disponible contra el sistema de muestreo desarrollado.	70

INDICE DE FIGURAS

Figura 1.1	Representación del estado del arte de convertidores analógico – digital.	4
Figura 2.1	Esquema típico de un modulador delta-sigma de un bit.	8
Figura 2.2	Representación de la amplitud de una señal en una modulación por densidad de pulso.	9
Figura 2.3	Esquema completo de un convertidor delta-sigma de un bit.	11
Figura 3.1	Localización de las instalaciones del Centro Nacional de Metrología, lugar donde fue realizada la presente investigación.	18
Figura 3.2	Ventana Hanning de 1024 puntos.	25
Figura 3.3	Minimización del efecto de corrimiento en la determinación del valor RMS debido a muestreo no síncrono.	26
Figura 3.4	Diagrama a bloques del sistema de muestreo de alta resolución desarrollado.	31
Figura 3.5	Grafica de estabilidad a largo plazo para el valor de 10 V de un diodo Zener Fluke 732B.	33
Figura 3.6	Diseño esquemático del divisor construido para ser utilizado como fuente de referencia de $\pm 2.5V$.	34
Figura 3.7	Divisor 4:1 utilizado como fuente de referencia de $\pm 2.5 V$.	34
Figura 3.8	Arquitectura básica para la implementación de un acumulador de fase, el cual es la base del reloj maestro de 4.096 MHz para el	38

	convertidor analógico-digital.	
Figura 3.9	Diagrama esquemático del amplificador de entrada utilizado en el sistema de muestreo.	40
Figura 3.10	Amplificador de entrada completamente diferencial montado sobre la tarjeta de circuito impreso.	40
Figura 3.11	Diagrama de tiempo de la interfaz SPI del ADS1281.	45
Figura 3.12	Diagrama de bloques del control del ADC por parte del FPGA.	46
Figura 3.13	Algoritmo para la determinación del valor rms de una señal periódica de tensión eléctrica continua.	49
Figura 3.14	Arquitectura implementada en el FPGA para resolver matemáticamente el algoritmo; diseño de la MAC que resuelve la multiplicación de alta resolución.	50
Figura 4.1	Diseño implementado de tarjeta de circuito impreso para el convertidor analógico-digital.	54
Figura 4.2	Sistema de muestreo basado en un convertidor analógico-digital Δ - Σ y un sistema FPGA Xilinx Spartan 3A-DSP 1800A.	55
Figura 4.3	Fotografía tomada durante el proceso de caracterización del sistema de muestreo digital.	55
Figura 4.4	Proceso de caracterización de la fuente de referencia de 2.5 V.	57
Figura 4.5	Análisis espectral utilizando la transformada rápida de Fourier y la ventana Hanning de	58

	1024 puntos.	
Figura 4.6	Formas de onda medidas en la entrada del circuito aislador H.P. HCPL-091J.	59
Figura 4.7	Resultado de la caracterización del amplificador de instrumentación completamente diferencial utilizando el algoritmo Swerlein.	61
Figura 4.8	Resultado de la caracterización del amplificador de instrumentación completamente diferencial utilizando el algoritmo Swerlein para linealidad utilizando la ganancia del amplificador como parámetro.	62
Figura 4.9	Efecto de frecuencia en el amplificador completamente diferencial.	63
Figura 4.10	Medición de tensión eléctrica continua a 1.018 V.	68
Figura 4.11	Medición de tensión eléctrica alterna a 517 mV a 60 Hz, muestra el desempeño con una desviación estándar menor a ± 695 nV.	68
Figura 4.12	Medición desde 15 V a 120 V a 60 Hz, muestra incertidumbre tipo A para dos tiempos de promediación distintos.	69
Figura 4.13	Calibración desde 15 V a 120 V a 60 Hz, compara 2 instrumentos comerciales contra el sistema de muestreo desarrollado.	71
Figura 4.14	Comparación del tiempo de procesamiento contra la incertidumbre tipo A obtenidos en una medición de tensión eléctrica alterna a 120 V y 60 Hz.	72

1. INTRODUCCIÓN.

Los convertidores analógico-digitales se han utilizado en aplicaciones de metrología eléctrica por más de 30 años, siendo el corazón de medidores digitales de tensión eléctrica. El propósito de cualquier convertidor analógico-digital es obtener una representación digital de una señal analógica, típicamente, tensión eléctrica. Normalmente la etapa de medición de un instrumento de medición de variables eléctricas tiene dos bloques principales: 1) un sistema de control y 2) un sistema de medición en sí mismo, el cual requiere de constantes verificaciones, ajustes y calibraciones. Este sistema de medición es en realidad un sistema de información, es decir, es un sistema que reúne información del fenómeno a medir en su entrada y a la salida del mismo se tiene información útil sobre alguna de las propiedades físicas del fenómeno de interés.

El propósito de un sistema de información es sintetizar aquellos datos que son útiles al usuario, el que usualmente no está interesado en lo que sucede en el interior del sistema de información. Sin embargo, la definición de un sistema de medición es más amplia, el propósito de todo sistema de medición es extraer información sobre un fenómeno físico. La información se extrae con los instrumentos de medición utilizando un proceso adecuado. Al final, el resultado de medición se obtendrá a partir del análisis de la información que proporciona el sistema de medición.

Idealmente se requiere que el sistema de medición no contribuya en nada a la determinación de una medición, en otras palabras, debe ser un sistema no intrusivo y el cual no influirá en la determinación de la propiedad física del objeto bajo medición.

En la práctica, estos sistemas son muy difíciles de lograr ya que existen variables de influencia sobre el fenómeno que se está estudiando. Dichas perturbaciones deberán ser cuantificadas para poder relacionar de manera

razonable un resultado de medición con la propiedad específica de un fenómeno sin cometer errores adicionales. Existe un campo del conocimiento relativo a las mediciones el cual estudia todos los aspectos posibles de un sistema de medición, esta ciencia es conocida como metrología.

Fluke Corporation (1994) en su libro *Calibration: Phylosophy in practice*, define de manera simple a la metrología como la ciencia de las mediciones. La mayoría de las actividades que se hacen tiene que ver con mediciones. Diseñar, planear y analizar los resultados de un sistema de medición es el interés fundamental en la metrología.

Tradicionalmente los convertidores analógico-digitales utilizados en instrumentación de alta exactitud (por ejemplo multímetros de señales eléctricas de 8 ½ dígitos) son convertidores que van en intervalos de operación de ± 2 V a ± 20 V, hasta la década de los 90's estos convertidores típicamente eran de dos tipos: integradores y de aproximaciones sucesivas.

Como ejemplo, Ronald Swerlein (1991) propone usar el convertidor analógico-digital embebido en un multímetro de alta exactitud comercial para lograr niveles de exactitud de 0.001% en valores de señales de tensión eléctrica alterna que van de 10 mV a 700 V con frecuencias de 0.1 Hz a 1 kHz. Durante más de una década esta solución ha sido base para mediciones de alta exactitud en magnitudes alterna para muchos laboratorios nacionales de metrología alrededor del mundo como lo muestra Kyriazis (2002, 2003).

Kürten (2005) presenta un análisis de caso para el uso de un multímetro de 8 ½ dígitos utilizado como un sistema de muestreo de alta resolución (basado en un convertidor analógico–digital de aproximaciones sucesivas de 28 bits). En dicho estudio se muestra el resultado del desempeño del sistema de muestreo en conjunto con técnicas de Transformada Discreta de Fourier así como modelos validados para la medición de señales de tensión eléctrica continua y de

estimación de incertidumbre. Este tipo de sistemas es arquitectura de sistemas de muestreo utilizando un multímetro de 8 ½ dígitos para mediciones de señales alternas de baja frecuencia (menor a 1 kHz) es ampliamente utilizado en el ámbito de la metrología eléctrica de alta exactitud.

Sin embargo la aparición de una nueva generación de convertidores analógico-digitales como los conocidos como delta-sigma (Δ - Σ) de alta resolución ha disminuido la dependencia de sistemas basados en instrumentación comercial. Georgeakopoulos *et al* (2004) reportaron resultados de medición con incertidumbres menores a 1×10^{-6} para mediciones de tensión continua y mediciones eléctricas de baja frecuencia, usando un ADC de 24 bits. Estos resultados son especialmente buenos, sin embargo la señal de tensión de referencia utiliza un efecto Josephson, el cual es la realización física de la unidad de tensión eléctrica, el volt. Se espera que estos desarrollos sean empleados para mediciones de energía eléctrica de muy alta exactitud en un plazo de 3 años.

El estado del arte de los convertidores analógico – digital, se puede resumir en la figura 1.1, la cual muestra una relación de tasa de muestreo dada en unidades de frecuencia contra el número de bits disponibles, resaltando aplicaciones típicas de esta relación. Cabe señalar que esta relación está exenta de aquellas técnicas de muestreo donde se utilizan arquitecturas de multi-núcleo con más de un convertidor analógico – digital o bien técnicas utilizando un solo convertidor y ráfagas de muestreo con fase distinta de cero.

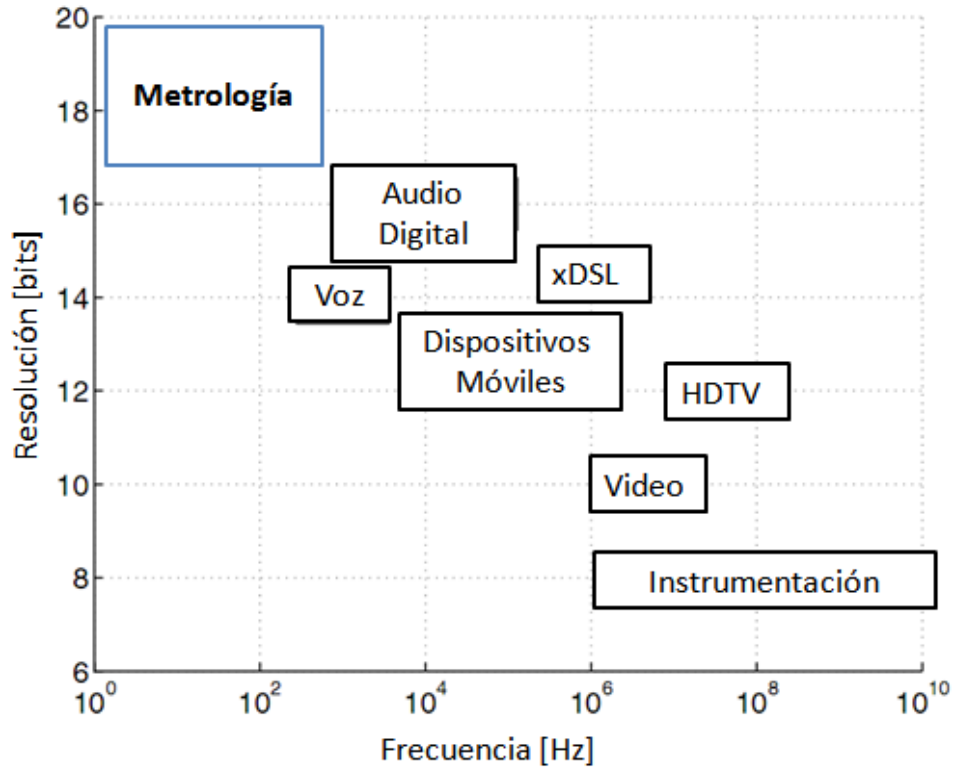


Figura 1.1. Representación del estado del arte de convertidores analógico – digital.

1.1 Objetivo.

La evaluación de los parámetros de desempeño (linealidad, resolución efectiva, nivel de ruido, etc.) de un convertidor analógico-digital comercial con tecnología Δ - Σ para determinar su aplicabilidad en sistemas de alta exactitud en sistemas de medición de tensión eléctrica de baja frecuencia con una exactitud mejor a 0.001 %.

1.2 Hipótesis.

La evaluación de los parámetros de desempeño en un ADC de alta resolución con tecnología Δ - Σ utilizando técnicas de procesamiento digital de

señales (DSP) permitirá decidir si el convertidor y el prototipo de sistema de muestreo propuesto son validos para aplicaciones de metrología eléctrica para mediciones de alta exactitud de tensión eléctrica alterna con incertidumbre menor a 0.001 % en un intervalo de 100 mV a 700 mV en frecuencias de DC a 60 Hz.

1.3 Reto del proyecto.

El grupo de tensión y corriente alterna del CENAM ha decidido abrir una línea de investigación que contemple la evaluación de los parámetros de desempeño de un convertidor analógico-digital delta sigma de muy alta resolución, para determinar su aplicación en sistemas de mediciones eléctricas de alta exactitud.

En este proyecto de tesis se investigaron los parámetros de desempeño de un convertidor analógico-digital de 31 bits, de tecnología delta-sigma, de Texas Instruments. El Centro Nacional de Metrología y la Universidad Autónoma de Querétaro no promueven el uso de esta marca comercial, ni de ninguna otra marca que pueda ser mencionada en este documento. Tampoco promueve que tales marcas o dispositivos sean los mejores para las aplicaciones que se discuten en este documento.

Entre otros aspectos, en este proyecto es necesario investigar el desempeño del convertidor analógico-digital de alta resolución en:

- No linealidad integral (INL), cuya especificación técnica indica ser menor a 1 parte en 10^6 .
- La resolución efectiva respecto de la tasa de muestreo..
- Efecto de señales electromagnéticas en el convertidor analógico-digital.
- La relación señal a ruido asociada con las referencias internas del convertidor y con la etapa de entrada del convertidor mismo.

1.4 Justificación.

Es necesario desarrollar mejores patrones de medición para el país y superar limitaciones tecnológicas: por ejemplo el multímetro digital HP/Agilent 3458A, que ofrece alta resolución, estabilidad y linealidad, pero resulta muy lento para aplicaciones de metrología eléctrica usando técnicas de procesamiento digital de señales.

Es necesario iniciar en el CENAM una línea de investigación en convertidores de datos con tecnología delta-sigma de alta resolución, pues la siguiente generación de medidores de magnitudes eléctricas de alta exactitud estará basada en esta tecnología.

Y por último, pero no menos importante, dentro de las funciones del Centro Nacional de Metrología, está enmarcada la formación de recursos humanos especializados para el cumplimiento de las funciones propias del Centro, dicho en otras palabras, fortalecer el capital humano en investigación y desarrollo tecnológico para satisfacer necesidades presentes y futuras en metrología, de la industria, la ciencia y las actividades económicas nacionales relacionadas con mediciones.

2. REVISIÓN BIBLIOGRÁFICA.

En este capítulo se describe de manera extensa la revisión bibliográfica hecha en el desarrollo del proyecto, esencialmente se describen 3 temas principales:

1. Los convertidores analógico-digitales con tecnología Δ - Σ .
2. Sistemas de muestreo orientados a aplicaciones de metrología eléctrica.
3. Estimación de valores RMS a partir de muestreo no síncrono.

2.1 Convertidores analógico-digitales con tecnología Δ - Σ .

Según Bryant (2000), los convertidores analógico-digitales de tecnología delta-sigma han sido conocidos por cerca de 30 años, pero recientemente se ha tenido la tecnología (VLSI) para fabricarlos en circuitos integrados monolíticos de bajo costo. Ahora son utilizados en una variedad muy grande de aplicaciones que requieran bajo costo, bajo ancho de banda, bajo consumo de potencia, y alta resolución en conversiones analógico-digital.

Un convertidor delta-sigma contiene electrónica analógica muy simple (un comparador, una referencia de tensión, un switch y uno o más integradores y sumadores, estos últimos implementados de manera analógica) y una etapa digital compleja. La etapa digital consiste en un procesador digital de señales el cual actúa como filtro (generalmente, pero no invariablemente, un filtro paso bajo). Para poder entender el principio básico de funcionamiento de un convertidor delta-sigma se debe estar altamente familiarizado con términos como sobremuestreo, ruido de digitalización, filtrado digital y decimación digital.

Las técnicas de modulación Sigma-Delta han sido usadas satisfactoriamente en las aplicaciones que requieren conversión análogo digital, en las últimas dos décadas. Aunque los conceptos de Sigma-Delta existen desde

la mitad del siglo, solo con los recientes avances en tecnologías VLSI ha sido posible darle un manejo adecuado al flujo de bits generado por el ADC de 1 bit.

La operación básica de un convertidor Delta-Sigma es intercambiar tasa de muestreo por resolución. Las señales son muestreadas a una tasa mucho mayor que la de Nyquist, generalmente se emplea un solo bit de resolución en la conversión analógica a digital.

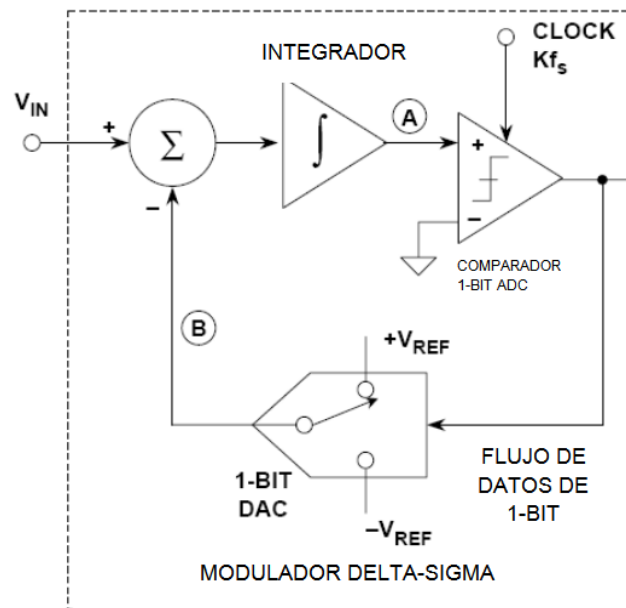


Figura 2.1. Esquema típico de un modulador delta-sigma de un bit (Tomado de Mixed-Signal and DSP Design Techniques, publicado por Analog Devices en el año 2000, página 3-12).

La figura 2.1 muestra una configuración de modulación Δ - Σ . La señal analógica de entrada V_{IN} se resta respecto del flujo de bits (bitstream), cuya densidad, es una representación de la amplitud de la señal analógica, el cual se presenta en la figura 2.2.

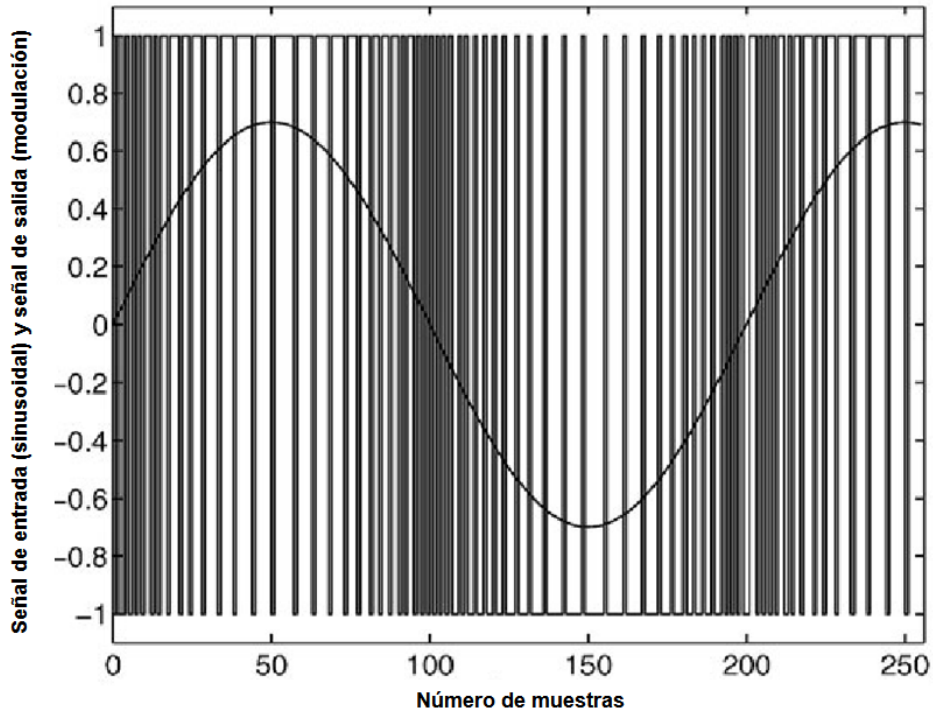


Figura 2.2. Representación de la amplitud de una señal en una modulación por densidad de pulso, se muestran las señales de entradas y salida a un modulador de primer orden Δ - Σ .

El error a la salida del diferenciador se integra y se presenta a un comparador, cuya salida tiene dos valores: 0 o 1, dependiendo si la salida del integrador es mayor o menor que la tensión de referencia del comparador.

La forma más simple de entender la operación es suponer una pequeña variación de una condición en estado estable. Por ejemplo, suponer una entrada analógica positiva y el promedio del “bitstream” es alto comparado con la entrada analógica, entonces la salida del sumador será negativa. La salida del integrador hará que el comparador genere un cero a su salida. Es claro que el promedio del “bitstream” debe seguir de cerca la forma de onda de la señal análoga de entrada.

El comparador es simplemente un amplificador con muy alta ganancia, el cual tendrá salidas de 1 o 0, dependiendo de la diferencia entre el valor del integrador y de la tensión de referencia. Esta configuración forma un lazo de

retroalimentación negativa de alta ganancia, y se espera que un convertidor Sigma- Delta tenga una excelente linealidad y un error muy pequeño, además minimizar los efectos del deterioro de los componentes con el tiempo.

En la modulación Delta-Sigma, la integración de la señal del diferenciador es codificada en lugar de la propia señal, la modulación puede ser entendida como un modulador Delta convencional precedido de una malla de integración. En la modulación Delta la señal analógica es aproximada con series de segmentos, cada segmento de la señal aproximada es comparado con la señal original para determinar si crece o decrece en amplitud; el valor de los siguientes bits está determinado por esta comparación, y solo los cambios de información son transmitidos, es decir, si no existe cambio en la comparación se seguirá transmitiendo el mismo valor (0 o 1).

El modulador Delta-Sigma descrito hasta ahora es esencialmente un ADC con solo un bit de resolución. La resolución puede ser aumentada promediando la salida del modulador, lo cual se puede lograr a través de un filtro pasabajos, seguido de un decimador el cual convierte la señal de un bit en una señal de N bits, en otras palabras para obtener alta resolución es necesario contar con una etapa de procesamiento digital de la señal de un bit, como lo muestra la figura 2.3.

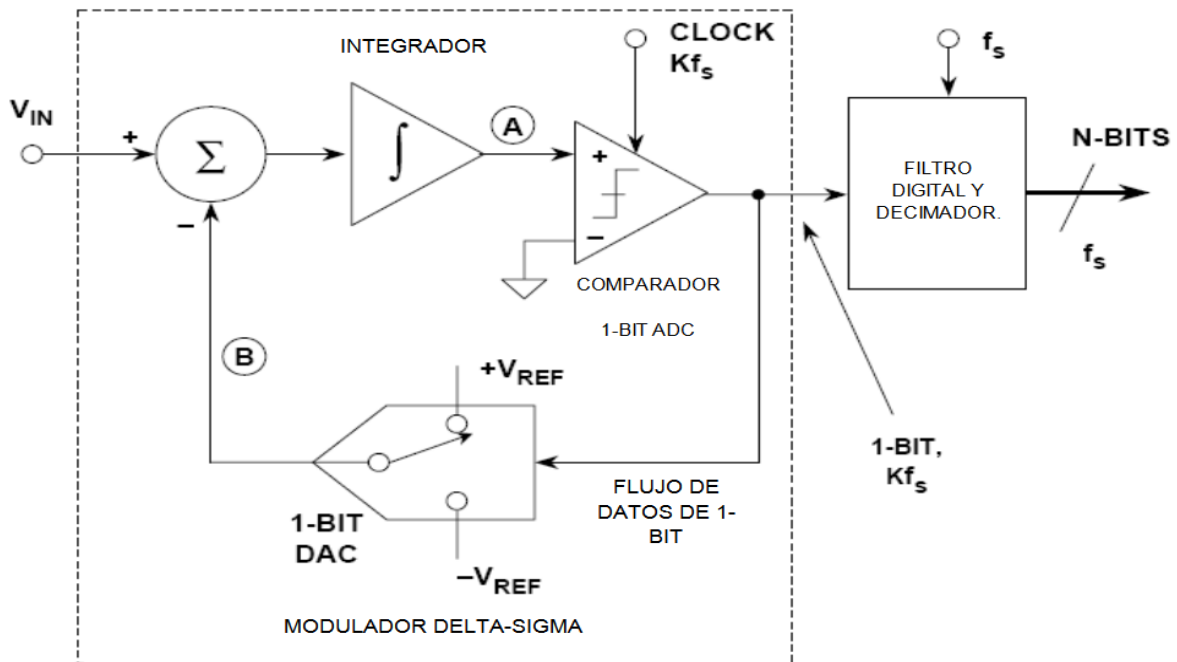


Figura 2.3. Esquema completo de un convertidor delta-sigma de un bit (Tomado de Mixed-Signal and DSP Design Techniques, publicado por Analog Devices en el año 2000, página 3-12).

En el presente proyecto se estudiará un convertidor analógico – digital, ADC, del tipo Δ - Σ , de 31 bits de resolución. Según sus especificaciones técnicas, este ADC dispone de un modulador de alta exactitud, de cuarto orden, con dos bloques de modulación, entre los cuales es posible estudiar la capacidad del convertidor en el primer bloque para minimizar el ruido de cuantización que es característico del procesamiento delta-sigma. Por su parte, el segundo bloque del modulador está diseñado para disminuir el ruido de cuantización que el primer bloque no pueda discriminar, por lo que también es objeto de estudio en este proyecto.

El convertidor analógico-digital elegido para llevar a cabo esta investigación es el modelo ADS1281 de Texas Instruments, el cual es un convertidor de sobremuestreo Δ - Σ que opera a una frecuencia nominal de 4.096 MHz máximo. Dentro del mismo encapsulado cuenta con etapas de filtrado digital

y de decimación para un intervalo operativo de 250 muestras por segundo a 4000 muestras por segundo, cuenta con una resolución nominal de 31 bits y su especificación más importante es una no-linealidad integral de $\pm 0.6 \mu\text{V/V}$.

El diseño y fabricación de la tarjeta de circuito impreso (PCB por sus siglas en inglés) que contiene el convertidor analógico-digital toma en cuenta técnicas para la separación de tierras digitales y tierras analógicas con el fin de reducir posibles contribuciones indeseables, como capacitancias parasitas, inductancia en, señales inducidas por campos electromagnéticos de manera conducida o radiada.

Con el fin de obtener el máximo desempeño del ADS1281, fue necesario implementar algunas de las mejores técnicas y prácticas de diseño electrónico. Existen aspectos críticos como lo son: puesta a tierra, el diseño del plano de tierra, diseño del plano de potencia, aislamiento digital, estabilidad en la señal de frecuencia en el reloj maestro y lo más importante, prestar especial cuidado en el diseño y fabricación de la fuente de tensión eléctrica de referencia. Los planos de potencia para señales analógicas y para señales digitales están separados y bien definidos en el diseño de capas del PCB. La clave para obtener un sistema de muestreo exitoso de alta resolución es mantener los retornos de tierra digital alejados de la entrada analógica a muestrear. Practicas de desacoplo son obligatorias.

De acuerdo con las especificaciones del ADS1281 para 1000 muestras por segundo, se pueden obtener un desempeño de -120 dB para una señal de 100 mV a 31.25 Hz y -121 dB para señales de tensión eléctrica continua. -120 dB de SNR (Relación señal a ruido por sus siglas en inglés) significa un nivel de ruido de $\pm 1 \mu\text{V/V}$, esto representa una resolución efectiva de hasta 20 bits a plena escala de 1.25 V.

2.2 Sistemas de muestreo en aplicaciones de metrología eléctrica.

En 1991, Swerlein R. propone usar instrumentación comercial para lograr niveles de exactitud de 0.001% en valores de 10 mV a 700 V con frecuencias de 0.1 Hz a 1 kHz. Durante años esta solución ha sido base para muchos laboratorios nacionales de metrología alrededor del mundo (Kyriazis 2002,2003).

El desarrollo de patrones de potencia en alterna es una clara tendencia en el campo de la metrología eléctrica moderna (Palafox 2007 y Kürten 2008) estos sistemas esta caracterizados en convertidores ADC de alta resolución (Iuzzolino 2009), tratando de responder al esfuerzo global por tener cada vez patrones de medición más exactos con modelos de incertidumbre

En esta década han existidos avances significativos en el diseño, construcción y caracterización de sistemas metrológicos en los campos de potencia y energía así como en calidad de la energía, por ejemplo Tóth (2005) propone un sistema de calibración de potencia puede alcanzar incertidumbre de alrededor de 30 $\mu\text{W/W}$, otro ejemplo descrito por Palafox et al (2007) donde se consiguieron incertidumbres del orden de partes por millón usando arquitecturas de ADC multi-núcleo. Este tipo de desarrollos son la nueva generación de sistemas de medición de magnitudes alternas, los cuales son utilizados como sistemas de transferencia o bien de referencia en los laboratorios de metrología más prestigiosos en el mundo.

Se han hecho estudios (Kürten et al 2005) sobre tecnología comercial, donde los multímetros de alta exactitud soportan convertidores analógico-digital de 28 bits resultando ideales para mediciones de señales sinusoidales. En 2009, Iuzzolino, Palafox, Kürten, Mohns y Brendel publican los resultados del diseño y caracterización de un sistema de muestreo analógico – digital $\Delta-\Sigma$ en aplicaciones de metrología eléctrica de alta exactitud. Usando un ADC $\Delta-\Sigma$ comercial de 24 bits (Modelo AD7763), dicha publicación es sumamente importante por dos

cuestiones, primero, se diseña y caracteriza un sistema de muestreo usando tecnología comercial y segundo lo realizan investigadores del PTB de Alemania, el cual es instituto nacional de metrología en Alemania, es decir, el equivalente del CENAM. Esto solo demuestra que existe una tendencia internacional a buscar sistemas de referencia cada vez más exactos para mediciones electromagnéticas, principalmente en el campo de corriente continua y baja frecuencia. El diseño presentado se asemeja a la propuesta que se realiza en el presente trabajo, es decir, cuentan con una referencia de tensión eléctrica continua de bajo corrimiento y bajo ruido, etapa digital montada sobre CPLDs, aislamiento óptico para las etapas analógicas y digital así como un convertidor comercial, la principal diferencia está sustentada en el hecho que el convertidor que se utiliza en dicha publicación es utilizado como un bloque único (modulador-decimador digital), en la propuesta realizada en esta tesis si bien se utiliza un convertidor de la misma manera, como bloque, además se busca mejorar el desempeño a partir de un procesamiento digital de señales externo con el uso exclusivo del modulador Δ - Σ . Los experimentos desarrollados para la caracterización resultan ser una guía práctica para la evaluación de los parámetros de desempeño de un convertidor analógico – digital de alta resolución, la caracterización se realizó usando diferentes sistemas de referencia, desde el uso de un sintetizador con base en el efecto cuántico Josephson hasta termoconvertidores para la comparación de señales en valores RMS. Los experimentos de prueba propuestos en esta tesis serán en algunos casos similares. Los resultados más relevantes presentados en este trabajo son una resolución efectiva de al menos 19 bits en una relación de muestreo de 64 kHz y tal vez la más importante una no linealidad de 7 μ V/V en el intervalo de -4 V a 4V, dando una referencia estimada del desempeño de los convertidores.

En los sistemas digitales de mediciones eléctricas, el punto crítico es sin duda alguna el convertidor analógico-digital.

Desde que los convertidores delta-sigma están basados en técnicas de filtrado digital, cerca del 90% de los sistemas digitales son implementados en circuitos digitales monolíticos mejorando la compatibilidad entre los sistemas digitales. Existen ventajas adicionales como lo son: alta confiabilidad, incremento en la funcionalidad de los sistemas y el más importante es que se puede tener alta precisión en un chip de bajo costo. En el mercado existen convertidores delta sigma como el introducido por Texas Instruments (2007), el modelo 1281, que cuenta con un modulador delta-sigma de cuarto orden y una serie de algoritmos de filtrado digital pre-programados.

Algunos convertidores analógico-digital convencionales, como el convertidor de aproximaciones sucesivas o los convertidores de tipo flash, que operan a la tasa de conversión de Nyquist requieren de bloques adicionales que complican su implementación y uso, entre estos bloques están un complicado filtro paso-bajo analógico (comúnmente llamado filtro *anti-aliasing*) para limitar la máxima frecuencia de entrada del convertidor, un circuito de muestreo y retención. Por otro lado los convertidores delta-sigma usan convertidores analógico-digitales de baja resolución (resolución de 1 bit), una técnica llamada noise shaping, y una tasa de conversión muy alta (llamada sobremuestreo). La alta resolución puede ser conseguida con técnicas de decimación digital (reducción en la tasa de muestreo con técnicas digitales). Por esto y más razones son sumamente atractivos para la implementación en sistemas de adquisición de señales eléctricas, y su aplicación en instrumentos de mediciones eléctricas es muy prometedora.

2.3 Estimación de valores RMS a partir de muestreo no síncrono.

La medición de valores RMS de señales periódicas es común en sistemas basados en tarjetas de adquisición, basados en PC o bien en multímetros de mano como en multímetros de 8 ½ dígitos que forman parte de sistemas de referencia de mediciones de tensión eléctrica alterna.

Hoy en día es común encontrar convertidores analógico–digital (ADC) de muy alta resolución y exactitud para aplicaciones de medición de valores RMS en señales de baja frecuencia en laboratorios de metrología alrededor del mundo, Iuzzolino et ál (2009) muestran un sistema basado en un ADC de alta resolución que alcanza niveles de incertidumbre tipo A de $3 \mu\text{V/V}$ para señales de tensión RMS con niveles de hasta 2 V y frecuencias menores a 1 kHz, sin embargo no se da indicio del algoritmo utilizado para la determinación del valor RMS.

Existen propuestas de solución a la determinación del valor RMS basadas en metrología clásica (termoconvertidores multiunión) combinados con ADC – DAC de alta resolución, Pogliano et ál (2008), sin embargo este tipo de sistemas están orientados a la calibración de fuentes de tensión eléctrica alterna por medio de una comparación entre la potencia disipada en un calefactor por la señal alterna y una señal equivalente en corriente continua. Este tipo de instrumentos son poco prácticos en campo ya que la determinación de valores RMS requiere de tiempos considerablemente altos para realizar una única medición.

Existen propuestas en la determinación del valor RMS, utilizando ADC de 14 bits con resultados comparables a un equipo comercial Fluke 5790A dentro de 2×10^{-5} (Liu Yue et ál, 2008), sin embargo para garantizar estos resultados se deben cumplir 3 condiciones:

- 1) Muestreo síncrono con la señal fundamental a medir.
- 2) El periodo de medición debe ser igual a un múltiplo entero de ciclos de la fundamental de la señal de interés. La frecuencia de muestreo debe

ser igual a un múltiplo entero de la frecuencia fundamental de la señal de interés.

3) La frecuencia de la señal de interés debe ser conocida *a priori*.

Este tipo de sistemas resultan ideales para mediciones en tiempo real y en medidores auto-contenidos, donde la tensión eléctrica requiere medir la frecuencia de la fundamental con alta exactitud. La aportación principal de este trabajo es la implementación de una metodología para la caracterización de los parámetros de desempeño de un ADC Δ - Σ de alta resolución a partir de experimentación y validaciones con métodos de medición de referencia.

Normalmente la determinación del valor RMS (valor cuadrático medio por sus siglas en ingles) de una señal eléctrica se da a partir de sistemas de muestreo, los cuales tienden a ser incoherentes en las muestras que obtienen. Novotny y Sedlacek (2006) demuestran que los errores en la determinación del valor RMS de una señal periódica están directamente asociados con la coherencia de las muestras y con el algoritmo de procesamiento digital de señales seleccionado, así como a las fluctuaciones propias de la señal a medir.

Kürten et ál (2004) muestra que la medición del valor RMS de señales eléctricas periódicas se puede conseguir mediante un procesamiento matemático aplicado a una secuencia de muestras digitalizadas de la señal bajo observación siempre y cuando exista sincronía entre el muestreo digital y la frecuencia fundamental de la señal de interés.

Aun cuando se han hecho avances en la corrección de la fuga espectral y efectos de desvanecimiento espectral (*smearing*) debido a muestreo no sincrónico en la determinación del valor RMS de una señal (Hegedus et ál, 2010) la implementación de estos algoritmos no resultan ser simples en sistemas embebidos como el presentado en esta investigación, por lo que una aproximación utilizando algoritmos clásicos en el dominio del tiempo es elegida.

3. METODOLOGÍA.

Las pruebas de desempeño se realizaron a partir de mediciones de alta exactitud de distintos parámetros, permitiendo decidir cuál es la configuración más apropiada para la aplicación del convertidor en sistemas de medición de referencia en aplicaciones metrológicas de corriente continua y baja frecuencia. Dichas pruebas se realizaron en el laboratorio multifunciones y en el laboratorio de potencia y energía de la división de mediciones electromagnéticas del Centro Nacional de Metrología con sistemas de referencia que son mantenidos en el Centro, las condiciones ambientales en dichos laboratorios son controladas con una temperatura de $23\text{ }^{\circ}\text{C} \pm 1\text{ }^{\circ}\text{C}$, de humedad de $35\% \pm 15\%$ y línea de alimentación de 127 V a 60 Hz con distorsión armónica del menor a 0.1 %, en la figura 6 se muestra la localización del Centro Nacional de Metrología.



Figura 3.1. Localización de las instalaciones del Centro Nacional de Metrología, lugar donde fue realizada la presente investigación.

Los patrones e instrumentos necesarios para poder reproducir este trabajo de investigación se detallan en la tabla 3.1.

Tabla 3.1. Detalle de los instrumentos comerciales utilizados durante el presente trabajo de investigación.

Descripción del Instrumento	Magnitud de referencia	Puntos calibrados usados durante la investigación.
Calibrador multifunciones de alta exactitud Fluke 5720A.	Tensión eléctrica continua; tensión eléctrica alterna.	±2.5 V; (50 mV, 100 mV, 200 mV, 300 mV, 400 mV, 500 mV, 600 mV y 700 mV) a 50 Hz y 60 Hz, (15 V, 30 V, 60 V, 90 V y 120 V) a 50 y 60 Hz.
Multímetro de 8 ½ dígitos Hewlett-Packard 3458A.	Tensión eléctrica alterna	(50 mV, 100 mV, 200 mV, 300 mV, 400 mV, 500 mV, 600 mV y 700 mV) a 50 Hz y 60 Hz, (15 V, 30 V, 60 V, 90 V y 120 V) a 50 y 60 Hz.
Medidor de tensión eléctrica alterna Fluke 5790A.	Tensión eléctrica alterna	(15 V, 30 V, 60 V, 90 V y 120 V) a 50 y 60 Hz.
Dispositivo Zener Fluke 732B.	Tensión eléctrica continua.	1.018 V y 10 V.
Generador de onda arbitraria Agilent 33250A	Frecuencia.	Amplitud: 3.3 V. Frecuencia: 16 MHz.
Tarjeta de desarrollo FPGA Xilinx Spartan 3A-1800 DSP	N/A	N/A
Fuente de poder Tektronix PS280	Tensión eléctrica continua	± 12 V.

En el campo de la metrología eléctrica de alta exactitud en años recientes son cada vez más populares el uso de técnicas de procesamiento digital de

señales. Especialmente el muestreo de señales y su posterior procesamiento con distintas técnicas ha llegado a niveles de exactitud similares a los patrones y sistemas de referencia de la más alta exactitud comerciales a nivel internacional. La incansable búsqueda y la investigación de sistemas más exactos ha obligado a los institutos nacionales de metrología y mediciones (NMI's por las siglas en ingles National Metrology Institutes) alrededor del mundo a explorar este campo del conocimiento.

En el caso del presente proyecto se desarrolló una metodología clara y simple para el desarrollo de un prototipo para la evaluación de parámetros de desempeño de un sistema de medición en base a un convertidor analógico – digital delta – sigma (Δ - Σ) al cual de ahora en adelante llamaremos ADC Δ - Σ .

De manera general la metodología a seguir en el desarrollo del presente trabajo de tesis se soporta de dos líneas principales: el desarrollo de un sistema de muestreo del alta resolución en base a un convertidor analógico – digital delta – sigma (Δ - Σ) y el desarrollo, análisis e implementación de algoritmos de procesamiento digital para la determinación del valor RMS de una señal periódica embebido en FPGA.

El desarrollo, implementación, prueba y análisis de un sistema de medición, el cual es un sistema basado en un ADC Δ - Σ modelo 1281 de Texas Instruments. Dicho convertidor tiene embebido un modulador delta – sigma así como una etapa de filtrado y decimación digital preestablecida la cual cuenta con las siguientes características:

Una etapa de decimación digital, la cual reduce la relación de muestreo inicial, la cual es de 1,024 MHz para el modulador. Dicha etapa es un filtro síncrono de 5to orden con una función de transferencia dada por:

$$H(z) = \left[\frac{1 - z^N}{1 - z^{-1}} \right]^5 \quad (1)$$

Donde N, puede tomar los valores siguientes: {128, 64, 32, 16, 8}, con lo cual se puede modificar la relación de reducción de la tasa de muestreo hasta un mínimo de 8000 muestras por segundo.

Una etapa de filtrado de respuesta finita al impulso (FIR por sus siglas en ingles) el cual tiene una etapa de decimación que va hasta las 250 muestras por segundo.

Todas estas características le permiten en principio prometer un desempeño que tienen una relación señal a ruido de alrededor de 130 dB y un error por linealidad máximo de 0,6 $\mu\text{V/V}$, es decir de 6×10^{-7} Volts para 1 V.

Además de estas características el convertidor presentado fue elegido debido a dos cualidades que resaltan por encima de todo lo anterior, alta resolución (32 bits) y una relación señal a ruido por encima de los 120 dB.

Es importante señalar y resaltar que estos parámetros servirán para evaluar el desempeño del convertidor y así poder decidir la factibilidad de su uso en el desarrollo de sistemas de referencia en mediciones electromagnéticas de alta exactitud.

En la caracterización metrológica del convertidor ADC Δ - Σ , se implementaron un medidor de tensión eléctrica alterna basado en el convertidor. El desempeño como medidor de tensión eléctrica alterna es un indicador objetivo de las capacidades reales del ADC Δ - Σ . Para conseguir el mejor desempeño como medidor de tensión eléctrica alterna se optimizan: la etapa de entrada del medidor y la conexión eléctrica con el ADC Δ - Σ , la calidad de las referencias de tensión eléctrica continua, de las señales digitales de control y datos del ADC Δ - Σ . Otra etapa importante en la caracterización del ADC Δ - Σ es la confiabilidad del

algoritmo de medición de valor RMS, considerando la naturaleza de la digitalización del ADC Δ - Σ .

Se muestran los resultados al implementar técnicas de ventana sobre FPGA para la determinación del valor RMS de señales de baja frecuencia utilizando algoritmos clásicos en el dominio del tiempo además se presentan criterios de diseño, orientados a obtener incertidumbre tipo A de al menos ± 5 μ V/V para frecuencias de línea eléctrica.

3.1 Determinación del valor RMS de una señal periódica a partir de un método de ventana.

La determinación por métodos digitales del valor RMS de una señal eléctrica, es una aproximación a la definición del valor RMS, como se muestra en la siguiente expresión matemática:

$$V_{\text{RMS}} = \sqrt{\frac{1}{T} \int_{t_0}^{t_0+T} (f(t))^2 dt} \cong \sqrt{\frac{1}{N} \sum_{n=1}^{N-1} (f(n))^2} \quad (2)$$

V_{RMS} debe ser trazable a valores de patrones nacionales, esto se logra a través de la comparación de los resultados de medición del medidor desarrollado contra un sistema de referencia, el algoritmo Swerlein en conjunto con un sistema digitalizador basado en un multímetro H.P. 3458A, el cual es caracterizado con valores de medición conocidos.

En la segunda igualdad de la ecuación 2, se presenta una aproximación al valor RMS, la cual se puede obtener con métodos digitales, en donde N es el total de muestras adquiridas digitalmente de la señal bajo observación, y n es un índice sobre los mismos.

Los errores principales de medición al utilizar una aproximación como en la ecuación 2, es la diferencia en la energía de la señal bajo observación que se

medir con la implementación digital y la energía que se podría obtener si se usara la definición. Entre algunas causas de la diferencia de la energía medida, está la sincronización del sistema de muestreo digital con el inicio y fin del periodo de la señal $f(t)$ así como el muestreo de múltiplos exactos del periodo de la señal. Si se quiere alcanzar una incertidumbre de medición menor a $\pm 5 \mu V/V$ por medios digitales, es necesario minimizar los efectos por la falta de sincronización entre el muestreo digital y el periodo de la señal eléctrica.

Diferentes algoritmos se han propuesto para minimizar los efectos de un muestreo digital sincronizado (Kürten et ál, 2004 y Novotny y Sedlacek, 2006). En este trabajo se propone la implementación de un algoritmo matemático que se puede utilizar para la determinación del valor RMS de una señal eléctrica, utilizando medios digitales de muestreo y un procesador digital de señales basado en un FPGA. El algoritmo permite minimizar efectos:

- Del tamaño finito de las muestras digitales de la señal eléctrica bajo observación.
- Por falta de sincronización entre el inicio y fin del periodo de la señal eléctrica y la señal digitalizada. El mayor efecto es el truncamiento de la señal digitalizada respecto del inicio y fin del periodo de la señal eléctrica bajo observación, lo cual adiciona componentes de frecuencias a la señal digitalizada, los cuales no existen en la señal eléctrica original.
- Por la linealidad no-integral de un muestreador digital de alta resolución que se requiere para mediciones de valor RMS de alta exactitud.

La técnica de ventana para la determinación del valor RMS es un tema conocido en el área de procesamiento digital de señales, donde se utilizan las populares ventanas coseno elevado, definidas como:

$$w(n) = \sum_{i=0}^P a_i \cos\left(\frac{2\pi ni}{N}\right) \quad n = 1, 2, \dots, N-1 \quad (3)$$

Donde N es la longitud de la ventana, a_i son los coeficientes asociados a la ventana elegida, P es el orden de la ventana.

El valor RMS de una señal periódica pasada a través de una ventana coseno es:

$$V_{\text{RMS}}[w(n)*f(n)] = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} [w(n)*f(n)]^2} \quad (4)$$

El proceso de hacer pasar las muestras sin procesar a través de una ventana como la definida en la ecuación 3, cambia los valores de la señal a la que se desea determinar el valor RMS, provocando un claro desvío en dicha determinación. Para reducir este efecto indeseado es necesario multiplicar el valor RMS desviado por una constante que depende de la forma de la ventana elegida para la determinación del valor RMS. Dicha constante está dada por la ganancia normalizada de potencia de ruido (nnp g por sus siglas en inglés *normalized noise power gain*) mostrada por Salomon (1994) la cual se presenta en la ecuación (5).

$$\text{nnp g} = \frac{1}{N} \sum_{n=0}^{N-1} [w(n)]^2 \quad (5)$$

Por lo que la determinación del valor RMS de una señal periódica queda expresada en la ecuación 6:

$$V_{\text{RMS}}[w(n)*f(n)] = \sqrt{\text{nnp g}} * \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} [f(n)]^2} \quad (6)$$

La elección correcta de la ventana coseno permite reducir las posibles desviaciones derivadas de muestreo no síncrono, siendo esta la principal ventaja en el uso de un algoritmo en el dominio del tiempo para la determinación del valor RMS de una señal periódica.

La ventana Hanning fue elegida debido a la forma de la señal, tiene la ventaja que en sus extremos la función de la ventana se hace cero minimizando los errores debido a la no sincronía del muestreo, es decir se minimiza los efectos de muestrear en una fase distinta a cero, la relación exacta de esta función se muestra en la ecuación 7. En la figura 3.2 se puede observar la forma de esta ventana.

$$w(n) = 0.5 \left(1 - \cos \left(\frac{2\pi n}{N-1} \right) \right) \quad (7)$$

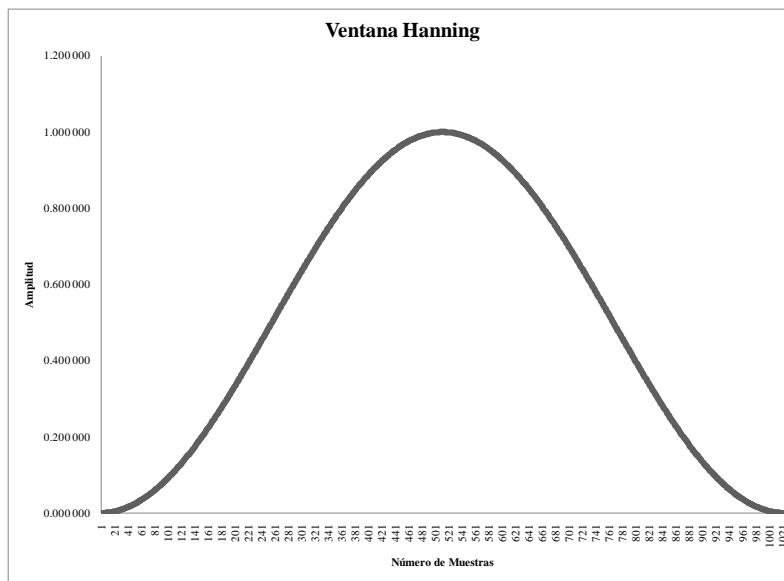


Figura 3.2 Ventana Hanning de 1024 puntos.

En la figura 3.3 se muestra el efecto de minimización que ocasiona el muestreo no síncrono en la determinación del valor RMS para una señal con amplitud 0.5 V.

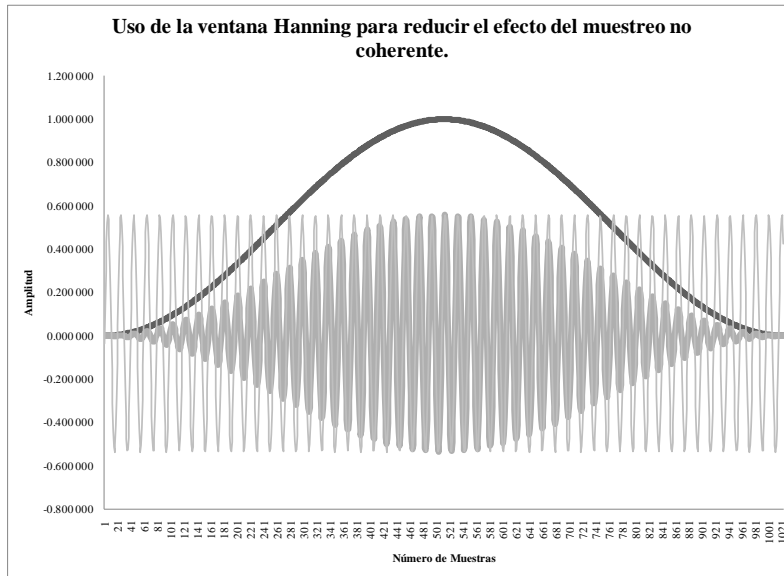


Figura 3.3 Minimización del efecto de truncamiento en la determinación del valor RMS debido a muestreo no sincrónico.

La determinación del valor RMS de una señal periódica utilizando la ventana Hanning está dada al combinar las ecuaciones 5, 6 y 7. Por lo que se puede considerar que dicha determinación es representada por la ecuación 8.

$$V_{\text{RMS}} [f(n)] \cong \sqrt{\frac{\frac{1}{N} \sum_{n=0}^{N-1} [w(n) * f(n)]^2}{0.375}} \quad (8)$$

3.2 Determinación del valor RMS de una señal periódica a partir del algoritmo Swerlein.

Para mediciones de valores rms para señales sinusoidales de baja frecuencia y de baja distorsión se han hecho estudios sobre la exactitud alcanzada utilizando un multímetro de 8 ½ dígitos Hewlett-Packard 3458A.

Swerlein (1991) describe un algoritmo sincrónico que permite alcanzar exactitudes muy superiores a las alcanzadas por el estado del arte de multímetros

e instrumentación de alta exactitud, particularmente para señales por debajo de 1 kHz.

La especificación de exactitud dada por el fabricante para el multímetro de 8 ½ dígitos para señales de tensión eléctrica alterna en el intervalo de frecuencias de 40 Hz a 1 kHz está en el orden de $\pm 100 \mu\text{V/V}$, con el algoritmo Swerlein se pueden alcanzar exactitudes del orden de $\pm 10 \mu\text{V/V}$.

Se basa en la definición clásica para la determinación de una señal periódica sinusoidal. La variante que propone es hacer una serie de ráfagas de muestreo con fase inicial 0° para la primera ráfaga y fase distinta de 0° para las siguientes, se computa el valor rms de cada grupo de muestras y el promedio de estas es la determinación del valor rms de la señal a medir.

Las capacidades del multímetro H.P. 3458A hacen posible la implementación de este algoritmo para señales no mayores a 1 kHz con repetibilidad en partes en 10^6 .

El algoritmo está basado en 3 características principales del multímetro de 8 ½ dígitos:

1. Capacidad para medición de la frecuencia de la señal que se desea medir.
2. Velocidades muy altas en el uso del convertidor analógico – digital SAR (aproximaciones sucesivas) con el que cuenta el multímetro.
3. La exactitud en la base de tiempo usada por el multímetro para tener muestras igualmente espaciadas en el tiempo.

Existen en este algoritmo dos fuentes de error principales, las cuales son corregidas dentro del programa de implementación propuesto por Swerlein (1991), el primero tiene que ver con el tiempo de apertura para cada muestra, es decir, se

requiere que el muestreo se haga con tiempos de apertura del convertidor analógico-digital muy reducidos provoca desviaciones en el cálculo del promedio de la señal, para tiempos de apertura muy grandes, la señal de entrada variara en el tiempo produciendo errores. En la implementación del algoritmo se propone tiempos de corrección de errores determinados experimentalmente.

Si el tiempo entre las muestras T_S multiplicado por el número de muestras n no es un número entero del período de la señal, aparecerá un error. Este error puede reducirse si n es muy grande, pero en el H.P. 3458A está limitado por la memoria interna (4000 muestras), para reducir este valor, el algoritmo propone desfazar ráfagas de muestras, mostrando que el valor promedio de seis ráfagas sucesivas e igualmente espaciadas elimina este error para la fundamental, señales sinusoidales del con distorsión por debajo del 0.1% permiten reducir a una contribución de partes en 10^6 a la incertidumbre del valor rms de la señal.

3.3 Normas internacionales para términos, definiciones y métodos de prueba de convertidores analógico-digital.

Basados en la norma *IEEE Standard for Terminology and Test Methods for Analog-Digital Converters (IEEE Std 1241-2010)*, la cual tiene como objetivo identificar las principales fuentes de error y provee de métodos de prueba normalizados que son útiles para la determinación del desempeño de un convertidor analógico – digital. El alcance de esta norma beneficia a fabricantes y a usuarios de los convertidores. Dentro de la norma hay un apartado para la descripción de especificaciones de nuevos productos así como para la comparación de productos existentes en el mercado.

Algunas de las pruebas descritas en la norma permiten corregir errores sistemáticos en los convertidores analógico – digital, por ejemplo, errores de ganancia, errores por offset entre otros.

La norma solo considera los convertidores analógico-digitales, que tengan valores de salida discretos en intervalos de tiempo discretos, por ejemplo, los convertidores que son cuantizados y muestreados. Más allá de que existen distintas topologías para la conversión del mundo analógico a una representación digital, la norma señala que la relación entre la señal de entrada y la salida digital en códigos binarios responde al principio en el cual la escala completa del convertidor analógico – digital es dividida en intervalos uniformes, con un ancho definido, el número de transiciones en el código binario está dada por la relación $2^N - 1$ donde N es el número de bits que puede resolver el convertidor.

Las especificaciones técnicas de un convertidor analógico – digital puede dividirse en dos campos: a) condiciones de prueba y b) resultado de prueba. En el primer grupo pueden numerarse las siguientes: temperatura, tensiones eléctricas de alimentación del circuito, tensiones eléctricas de referencia, frecuencia de reloj; en el segundo grupo están por ejemplo parámetros de desempeño del convertidor analógico – digital como lo son: disipación de potencia, número efectivo de bits, linealidad no integral, etc.

Las hojas de datos que los fabricantes de convertidores analógicos – digitales, especifican resultados de prueba asociados a condiciones de prueba muy específicos. La norma orienta las recomendaciones de pruebas para fabricantes y para usuarios por igual, sin embargo declara parámetros críticos de prueba para un convertidor analógico – digital

En la página 10 de la norma IEEE Std 1241-2010 se reproduce los parámetros críticos a evaluar dependiendo de la aplicación final del convertidor analógico – digital. Dicha tabla fue tomada de la.

En la tabla no se observa claramente donde puede estar ubicada una aplicación como lo es metrología, lo más cercano se encuentra en aplicaciones

típicas de adquisición de datos, donde los parámetros de desempeño claves son: No linealidad diferencial, No linealidad integral, offset, ruido, tiempo de respuesta, respuesta a escalón en escala completa. Estos parámetros sirven como guía de evaluación de los parámetros de desempeño estudiados en el presente trabajo de investigación.

3.4 Diseño del sistema de muestreo digital utilizando un ADC Δ - Σ .

El diseño propuesto consta de las siguientes etapas principales las cuales serán detalladas en extenso y se muestra como diagrama a bloques en la figura 3.4:

- Fuente de referencia de corriente directa (bajo ruido, baja deriva, bipolar).
- Convertidor analógico-digital Δ - Σ (Texas Instruments ADS1281).
- Aisladores Digitales.
- Reloj Maestro (4.096 MHz, basado en DDS).
- Amplificador de entrada completamente diferencial.
- Control y procesamiento de señales para mediciones de niveles de tensión eléctrica continua y tensión eléctrica alterna basado en un FPGA Xilinx XC3SD1800A.

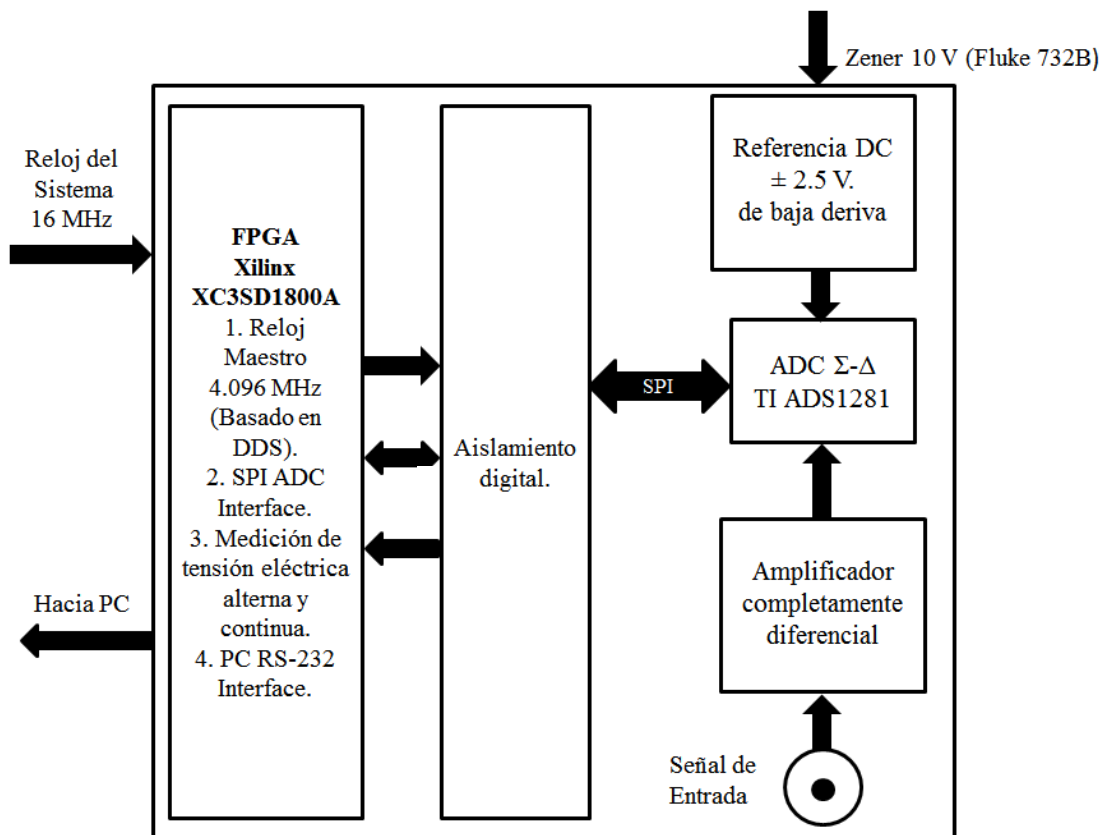


Figura 3.4. Diagrama a bloques del sistema de muestreo de alta resolución desarrollado.

3.5 Fuente de Referencia de corriente directa de ± 2.5 V.

Un desempeño pobre en el comportamiento de las referencias de tensión eléctrica continua de ± 2.5 V, enmascara el comportamiento real del ADC Δ - Σ . El nivel de ruido que contengan las referencias de tensión eléctrica continua se reflejara directamente en el comportamiento del medidor en tensión eléctrica alterna. Resultara en ruido en la medición del valor RMS de la señal de interés, sin que un posible mal desempeño del medidor de tensión eléctrica alterna este directamente asociado a un mal comportamiento del ADC Δ - Σ .

Siguiendo a Pickering (2004) y citándolo textualmente “La más grande fuente de error de un convertidor analógico – digital delta – sigma (Δ - Σ) es el ruido, estabilidad y la linealidad de la fuente de referencia de la retroalimentación DAC”, en otras palabras el corazón de un convertidor analógico–digital de alto desempeño es la estabilidad de la fuente de referencia, la calidad de un sistema de medición entonces estará determinada por su referencia de tensión eléctrica.

Esta unidad provee una tensión eléctrica de referencia altamente estable y exacta, el diseño está basado en un diodo Zener comercial, el Fluke 732B, el dispositivo de Zener entrega salida de 10 V y de 1.018 V, para los propósitos se utilizo la salida de 10 V, dicho Zener ha sido caracterizado en el tiempo garantizando una señal estable con niveles de incertidumbre por estabilidad a largo plazo (3 meses) menores a $\pm 0.4 \mu\text{V/V}$ ($4 \mu\text{V}$ para 10 V de forma absoluta) , lo cual garantiza estabilidades a corto plazo mucho menores a este valor. La figura 3.5 demuestra esto. El ADC Δ - Σ puede utilizarse para medir bajo estas condiciones señales de tensión eléctrica continua, resistencia eléctrica, corriente eléctrica continua, para mediciones de temperatura de contacto a partir de termopares o rtd's.

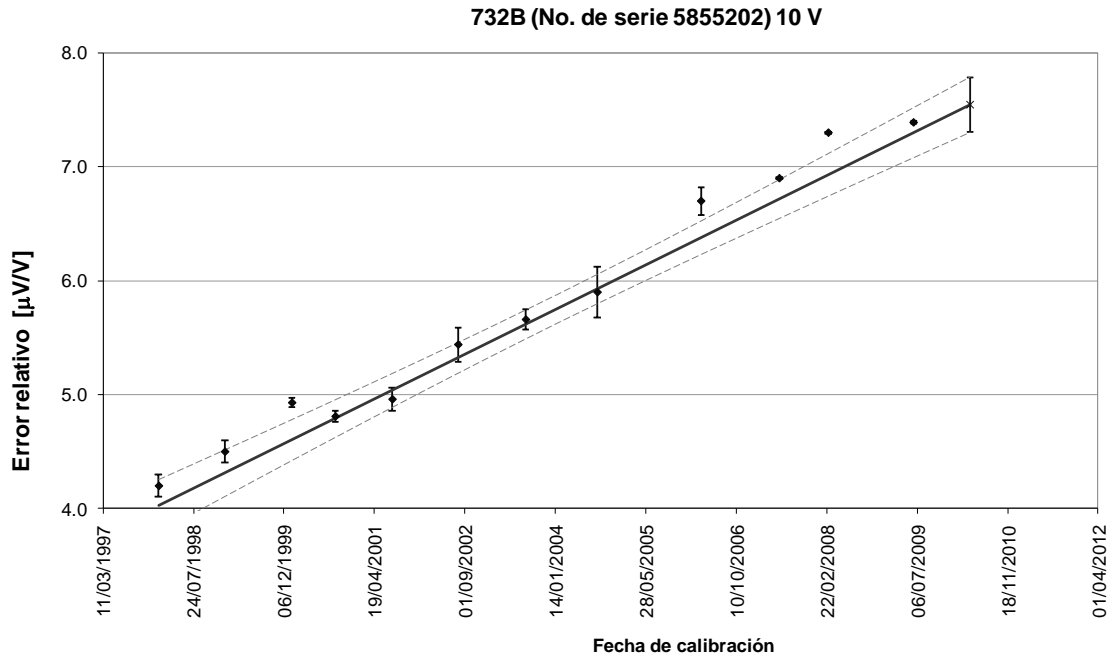


Figura 3.5 Grafica de estabilidad a largo plazo para el valor de 10 V de un diodo Zener Fluke 732B.

El diseño de la fuente de referencia de ± 2.5 V fue hecho en base a un divisor 4:1, se utilizaron resistencias de alta calidad marca vishay modelo VHP101 de 0.005 % de exactitud con un coeficiente térmico de $0.3 (\mu\Omega/\Omega)/^{\circ}\text{C}$. Con esto se garantiza que la estabilidad del divisor a largo plazo. En la figura 3.6 muestra el diseño esquemático del divisor y en la figura 3.7 se muestra la tarjeta de circuito impreso a partir de este.

Otro criterio de desempeño en el ADC $\Delta-\Sigma$ es su rechazo de modo común en las señales bajo medición. En lo relativo a las referencias de ± 2.5 V se buscó que no contuvieran ruido aditivo (por ruido conducido por cables en los circuitos de medición y por acoplamiento inductivo). Un criterio adicional de desempeño de esta etapa en particular, es la minimización de fuerzas electromotrices o tensiones eléctrica de origen termoeléctrico, es decir tensiones eléctricas que se presentan el contacto eléctrico del ADC $\Delta-\Sigma$ con el resto del circuito eléctrico. Estas fem's termoeléctricas se minimizan reduciendo los gradientes de temperatura en los contactos eléctricos entre dos metales.

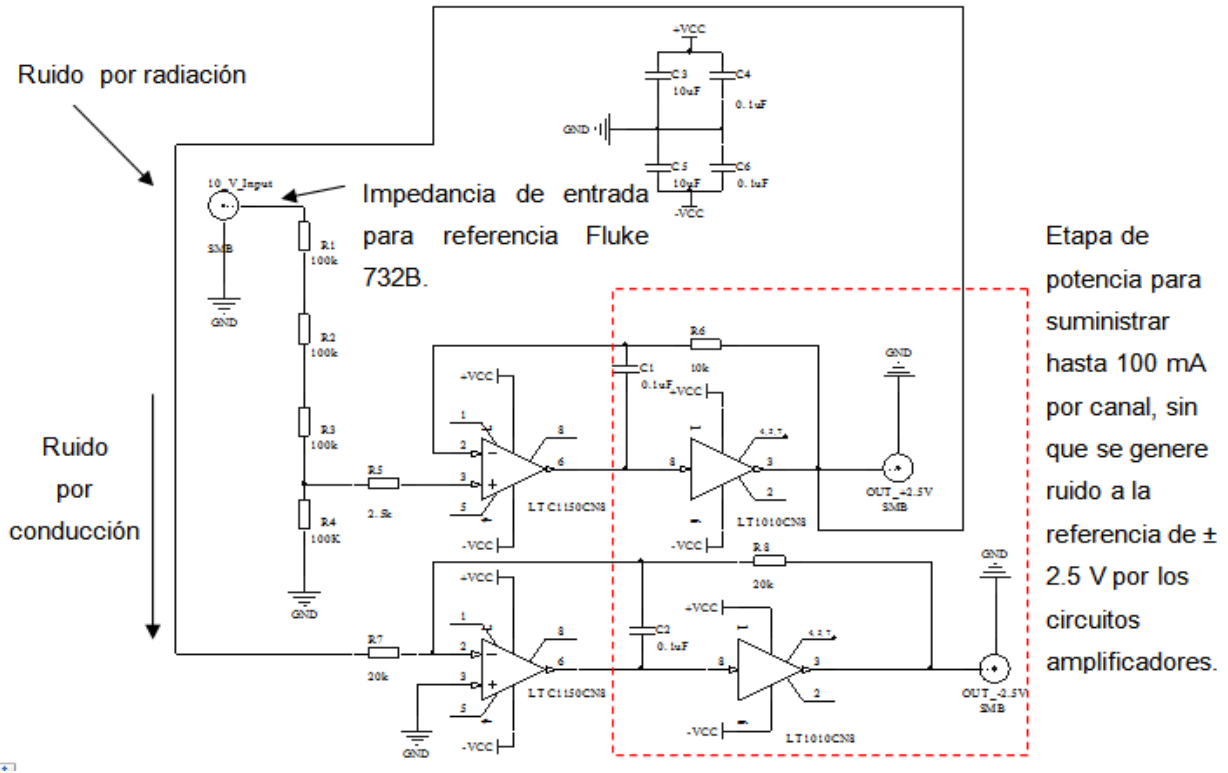


Figura 3.6 Diseño esquemático del divisor construido para ser utilizado como fuente de referencia de $\pm 2.5V$.

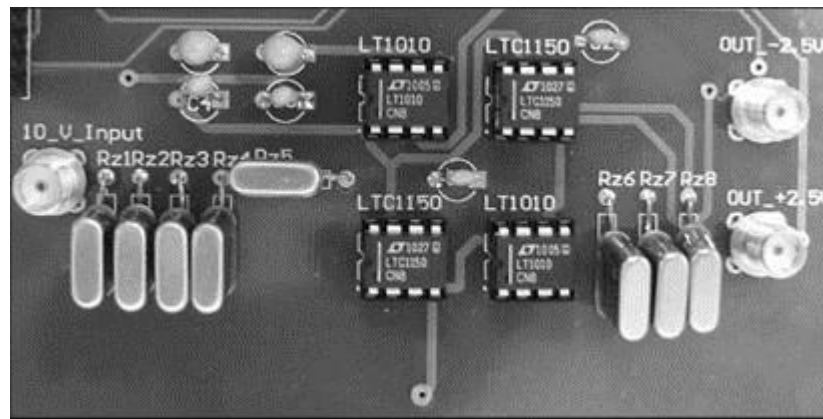


Figura 3.7 Divisor 4:1 utilizado como fuente de referencia de $\pm 2.5 V$.

La fuente de tensión de referencia fue calibrada contra un voltmetro digital de alta resolución, el cual usa un convertidor analógico-digital de 28 bits con tecnología de aproximaciones sucesivas, dicho instrumento ha sido caracterizado de manera extensiva con trazabilidad al patrón nacional de tensión eléctrica continua. Su estabilidad a corto plazo es mejor que $0.2 \mu\text{V/V}$ dentro de 24 horas, con una incertidumbre expandida de $\pm 0.7 \mu\text{V/V}$ la cual incluye, incertidumbre tipo A (desviación estándar de la medición) y la incertidumbre tipo B (Combinación de incertidumbre por estabilidad a largo plazo, resolución e incertidumbre de calibración).

3.6 Convertidor analógico-digital Δ - Σ (Texas Instruments ADS1281).

El convertidor analógico-digital (ADC) ADS1281 de Texas Instruments es un convertidor sobre muestreo que opera a una velocidad máxima de 4.096 MHz, en el mismo circuito se puede encontrar tasas de muestreo del orden de 250 muestras por segundo (MPS), 500 MPS, 1000 MPS y 4000 MPS, consistentes con 31 bits de resolución y la especificación de exactitud más importante: no linealidad integral del orden de $\pm 0.6 \mu\text{V/V}$.

El diseño de la tarjeta de circuito impreso (PCB por sus siglas en ingles) tomó en cuenta técnicas para el aseguramiento de compatibilidad electromagnética y protección de fenómenos de baja frecuencia como: Fem's termoeléctricas, ruido en referencias de tensión eléctrica continua, entre otros efectos sistemáticos indeseables. Dichas técnicas se enlistan a continuación:

1. Conexión de tierra adecuada para circuitería analógica y para circuitería digital. Conexión de un punto de referencia en un solo punto del circuito.
2. Retornos de corriente a tierra para cada circuito bien definido y sin conexión eléctrica con otras partes del PCB.

3. Reducción de pistas para evitar capacitancias parasitas e inductancias parasitas.
4. Blindaje eléctrico para evitar los efectos de campos electromagnéticos conducidos y/o radiados.

Para obtener el máximo desempeño del ADS 1281, es necesario implementar algunas de las mejores prácticas en diseño electrónico. Cuidado en el diseño de la alimentación de potencia, aislamiento digital para el procesador embebido en el FPGA en conexión con la circuitería del convertidor analógico-digital.

De acuerdo con las especificaciones de exactitud del convertidor analógico-digital, se pueden alcanzar desempeños de hasta -120 dB para niveles de tensión eléctrica alterna de 100 mV a 31.25 Hz y -121 dB para el mismo nivel en tensión eléctrica continua. -121.8 db de relación señal a ruido significa un nivel de ruido de $\pm 0.8 \mu\text{V}$, esto representa una resolución efectiva de hasta 21 bits a plena escala a $\pm 1.25 \text{ V}$.

El nivel de ruido en el sistema de muestreo no es exclusivo del nivel de ruido que el convertidor analógico-digital puede desarrollar por sí mismo (por ejemplo ruido de cuantización por modulación Δ - Σ , etc.), sino que es la suma de todos los componentes del mismo, el fabricante del convertidor no da datos suficientes para poder hacer una discriminación de dicho ruido. La evaluación de este se hizo utilizando análisis espectral con la transformada rápida de Fourier, y esta se llevó a cabo sobre la base del sistema de muestreo completo incluyendo el uso de la ventana Hanning, la cual fue elegida para la determinación del valor rms de una señal periódica, se compararon los resultados obtenidos a niveles superiores de tensión (700 mV) a 50 Hz, a 1000 muestras por segundo.

Se determino que la relación de muestreo optima entre desempeño y velocidad es de 1000 muestras por segundo, el fabricante del convertidor analógico-digital declara la mejor especificación de este a 250 muestras por segundo, pero limitaría al sistema de muestreo a medir señales por debajo de los 40 Hz, si se utilizara la configuración más rápida se tendría un comportamiento de -118 Db correspondiente a $\pm 1 \mu\text{V}$ en 1.25 V. Siendo elegida 1000 muestras por segundo, ya que tiene un comportamiento prometido en ruido de al menos de $\pm 0.5 \mu\text{V}$ en 1.25 V.

3.7 Aislamiento digital. Protección contra ruido radiado y conducido hacia el ADC Δ - Σ .

Los sistemas digitales por naturaleza son sistemas ruidosos, circuitos integrados con compuertas lógicas con consumos de potencia bajos parecieran en primera instancia que no son una fuente seria de ruido, sin embargo, las altas velocidades con las que cambian entre el valor lógico '1' y el valor lógico '0' combinado con la inductancia de los conductores que los interconecta, hacen que se produzca una de las fuentes de ruido más importantes en el diseño de un sistema digital. La tensión que se produce cuando la corriente cambia con el tiempo en un inductor está dada por:

$$V(t) = L \frac{di}{dt} \tag{9}$$

Henry Ott, en su libro "Noise Reduction Techniques in Electronic Systems" prueba que una sola compuerta que consume alrededor de 5 mA puede producir ruido con amplitud de hasta 1 V en sistemas con frecuencias en los MHz.

De lo anterior se concluye que un sistema con base en un FPGA como el que va a ser utilizado en el desarrollo del proyecto es una fuente importante de ruido, convirtiéndose en una necesidad primordial poder aislarla con un diseño de bajo ruido como el que se tiene con el convertidor analógico – digital.

3.8 Reloj maestro de 4.096 MHz.

Una fuente de error adicional está directamente relacionada con la calidad del reloj digital maestro de 4,096 MHz que necesita para operar el convertidor analógico-digital a su máxima capacidad de velocidad. En el presente trabajo se desea trabajar con un solo reloj maestro. La generación del reloj para el convertidor se realizara a partir de un algoritmo de generación de bases de tiempo con ciclo de trabajo del 50 %, el cual está embebido en el FPGA.

El algoritmo propuesto se basa en un DDS (Direct Digital Synthesizer por sus siglas en ingles) publicado por Romero-Troncoso *et al* (1999). Está basado en un acumulador de fase, que permite tener una estabilidad superior a la generada por una base de tiempo con contadores comunes. En la figura 3.8 se muestra la arquitectura básica de un acumulador de fase.

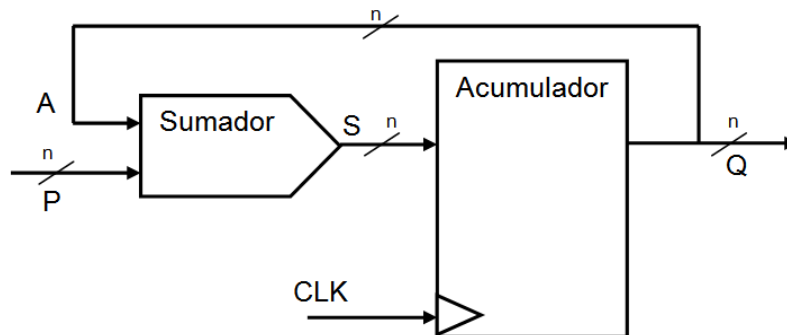


Figura 3.8. Arquitectura básica para la implementación de un acumulador de fase, el cual es la base del reloj maestro de 4.096 MHz para el convertidor analógico-digital.

El acumulador de fase es un sistema digital que genera una base de tiempo cuya frecuencia de salida sigue un patrón directamente proporcional a la

frecuencia base del reloj con el que se alimenta al FPGA. La frecuencia de este sistema está representada en la ecuación 10.

$$f_Q = \frac{P}{2^n} f_{CLK} \quad (10)$$

Donde f_Q es la frecuencia de salida del acumulador de fase, P es la fase o el factor proporcional a f_{CLK} y n es el número de bits utilizados para la aplicación. Para el diseño propuesto se utilizó un diseño de procesamiento por lotes (pipeline) de 5 etapas, para un total de 40 bits de operación y el reloj maestro es de 16 MHz, a partir del generador de onda arbitraria Agilent 33250A.

3.9 Amplificador de entrada completamente diferencial en el ADC Δ - Σ .

La baja impedancia de entrada del convertidor analógico-digital, 55 k Ω a 4.096 MHz obliga a que el sistema de muestreo tenga una etapa de acondicionamiento de señal. El convertidor cuenta con entradas para muestreo completamente diferenciales, por lo que el acoplamiento con el amplificador de entrada debe darse en estas condiciones.

Spinelli et al (2010) proponen un circuito de alta impedancia completamente diferencial, que utiliza un amplificador diferencial de entrada y otro amplificador diferencial de salida, para una aplicación en conjunto con un convertidor analógico-digital, el cual tiene entradas diferenciales, los resultados muestran sobresalientes en un ancho de banda de 17 Hz. En el presente trabajo de investigación se llevó las condiciones de operación de hasta 60 Hz. En la figura 3.9 se muestra el diagrama esquemático del amplificador de entrada del sistema de muestreo. En la figura 3.10 se muestra la tarjeta de circuito impreso

$$G = \left(1 + \frac{2R_2}{R_1}\right) * \left(-\frac{R_4 + R_5 + \frac{2R_4R_5}{R_6}}{R_3}\right) \quad (11)$$

Dichos valores son para $R_1 = 3.3 \text{ k}\Omega$, $R_2 = R_4 = R_5 = 1 \text{ k}\Omega$, $R_3 = 10 \text{ k}\Omega$ y $R_6 = 500 \text{ }\Omega$, y aplicando estos valores a la ecuación 11 se obtiene una ganancia de 0.963 636.

Los resistores seleccionados fueron de marca *Vishay* modelo Z201, los cuales tienen un excelente desempeño térmico, cuentan con un coeficiente de temperatura de $\pm 0.05 (\mu\Omega/\Omega)/^\circ\text{C}$, para ejemplificar esto, la máxima variación en el valor de un resistor de $1 \text{ k}\Omega$ con un gradiente de temperatura es de $50 \mu\Omega$, este desempeño garantiza una alta inmunidad a los efectos de temperatura en los valores de los resistores, la exactitud de estos por otro lado es de 0.005% , el criterio de diseño en este caso es el de mantener los valores de los resistores lo más próximo a su valor nominal, para que el desbalance que pueda existir en el amplificador completamente diferencial sea lo menor posible, minimizando efectos de desplazamiento de offset en la señal de interés a muestrear por el convertidor analógico-digital.

Los amplificadores operacionales seleccionados son el modelo AD8552 de la marca *Analog Devices*, el cual fue seleccionado debido a las características nominales ofrecidas por el fabricante contra el precio de mercado de este amplificador. Es un circuito doble que tiene tensiones de offset menores a $1 \mu\text{V}$, deriva por temperatura del orden de $5 \text{ nV}/^\circ\text{C}$, operación con una fuente de alimentación de una polaridad, y CMRR (Relación de Rechazo de Modo Común por sus siglas en inglés) del orden de 130 dB , esto supera los 124 dB prometidos por el convertidor analógico-digital permitiendo discriminar el efecto de ruido de fondo del amplificador en comparación con el ruido del convertidor analógico-digital.

El signo negativo de la ganancia significa una inversión de fase de la señal. Para los propósitos de esta investigación este no es un problema mayor, ya que es de interés de la misma caracterizar los parámetros de desempeño del convertidor y del sistema de muestreo en amplitud y no en fase.

El objetivo de la experimentación es proporcionar valores RMS de señales de referencia de bajo nivel (< 1 VRMS) y de baja frecuencia (60 Hz) trazables al patrón nacional de tensión eléctrica alterna, el cual es un termoconvertidor multi-uniones, el cual ha sido caracterizado contra los valores mantenidos en el PTB (Alemania, *Physikalisch-Technische Bundesanstalt*). Dicha trazabilidad no se directamente contra el termo-convertidor, se requiere un método de transferencia conocido, el método utilizado fue a través del algoritmo Swerlein y un multímetro H.P. 3458A usado en modo de muestreo.

Se realizaron experimentos para corroborar los siguientes parámetros del amplificador completamente diferencial:

1. Dispersión de mediciones utilizando desviación estándar convencional. Con la finalidad de determinar el nivel de ruido aportado por el amplificador de instrumentación, el experimento se realizó en condiciones de prueba de 100 mV a 700 mV rms en tensión eléctrica alterna a 60 Hz, utilizando para ello como fuente altamente estable el calibrador multifunciones Fluke 5720A y como medidor el multímetro H.P. 3458A así como el algoritmo Swerlein para la determinación de los valores rms de la señal de salida del amplificador de instrumentación. Este algoritmo propuesto por Swerlein (1991), investigador de la compañía Hewlett-Packard ha servido por años como sistema de referencia para la determinación de valores rms de señales de tensión eléctrica alterna de baja frecuencia (< 1 kHz) con alta exactitud, los reportes de los resultados indican exactitudes en plazos tan largos como un año menores a $10 \mu\text{V}/\text{V}$, esto resulta en exactitudes mejores que $10 \mu\text{V}$ durante calibraciones anuales en el

valor de tensión de 1 V. De este punto en adelante este algoritmo será indicado como Algoritmo Swerlein.

2. Linealidad en el intervalo descrito en el punto 1. A partir de las mediciones hechas para encontrar el nivel de ruido del amplificador se puede deducir con cierta facilidad el grado de no linealidad del mismo.

3.10 Control y procesamiento de señales para mediciones de niveles de tensión eléctrica continua y tensión eléctrica alterna basado en un FPGA.

Las arquitecturas presentadas en este trabajo de investigación se desarrollaron bajo un ambiente estándar en el lenguaje de síntesis y modelado de circuitos digitales conocido como VHDL, el criterio utilizado para esta selección fue el de utilizar un lenguaje estándar que permitiera la portabilidad de los diseños a más de un fabricantes de circuitos de lógica programable como lo son los FPGAs; existen en el mercado lenguajes de descripción de circuitos que suelen ser propios de una determinada herramienta o marca comercial, siendo posible la optimización de los diseños pero impidiendo la portabilidad a fabricantes distintos al seleccionado.

La primera etapa en el diseño del sistema digital embebido en FPGA es el control del convertidor analógico-digital, el convertidor cuenta con dos modos de configuración:

1. Modo Pin. A través de cambio en los pines del circuito integrado.
2. Modo Registro. A través del uso de una interfaz SPI dedicada y comandos particulares para la modificación de registros digitales internos.

Se eligió configurarlo a través del modo registro, debido a que los cambios de configuración que se pueden realizar de manera sencilla desde el programa embebido en el FPGA.

La segunda configuración importante en el convertidor analógico-digital es el modo de lectura de las muestras, tiene dos modos: carrera continua y por comando, el primero de ellos como su nombre lo indica inicia conversiones y no se detiene hasta que se da el comando stop, en el segundo modo se inicia la conversión a través de una indicación explícita a manera de modificación de un registro del convertidor analógico-digital deteniéndose cuando completa una muestra yendo a un estado de stanby, hasta que se reinicia el proceso nuevamente.

La interfaz del convertidor analógico-digital SPI, cuenta con cuatro señales fundamentales:

1. *DRDY* (flanco negativo). Señal de salida del convertidor analógico-digital, la cual indica cuando una nueva conversión esta lista para ser leída de manera serial.
2. *Dout*. Señal de salida del convertidor analógico-digital, es la línea física por donde se envía la información de conversión.
3. *Din*. Señal de entrada del convertidor analógico-digital, es la línea serial por donde se introducen los comandos de configuración del convertidor analógico-digital.
4. *SCLK*. Señal de entrada del convertidor analógico-digital, es la línea usada para introducir el reloj necesario para las operaciones de las líneas Din y Dout, la frecuencia seleccionada fue de 256 kHz.

En la figura 3.11 se muestra el diagrama de tiempos, de las señales involucradas en la interfaz SPI del convertidor analógico-digital para el modo de operación de lectura continua.

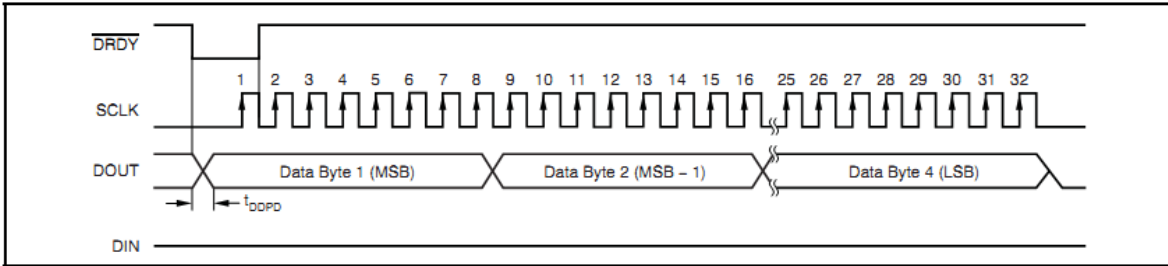


Figura 3.11. Diagrama de tiempo de la interfaz SPI del ADS1281.

Este diagrama de tiempos fue diseñado durante esta investigación y embebido en el FPGA seleccionado para el control y procesamiento de las muestras provenientes del convertidor analógico-digital. En la figura 3.12 se muestra la secuencia de operación del algoritmo de control del convertidor analógico digital.

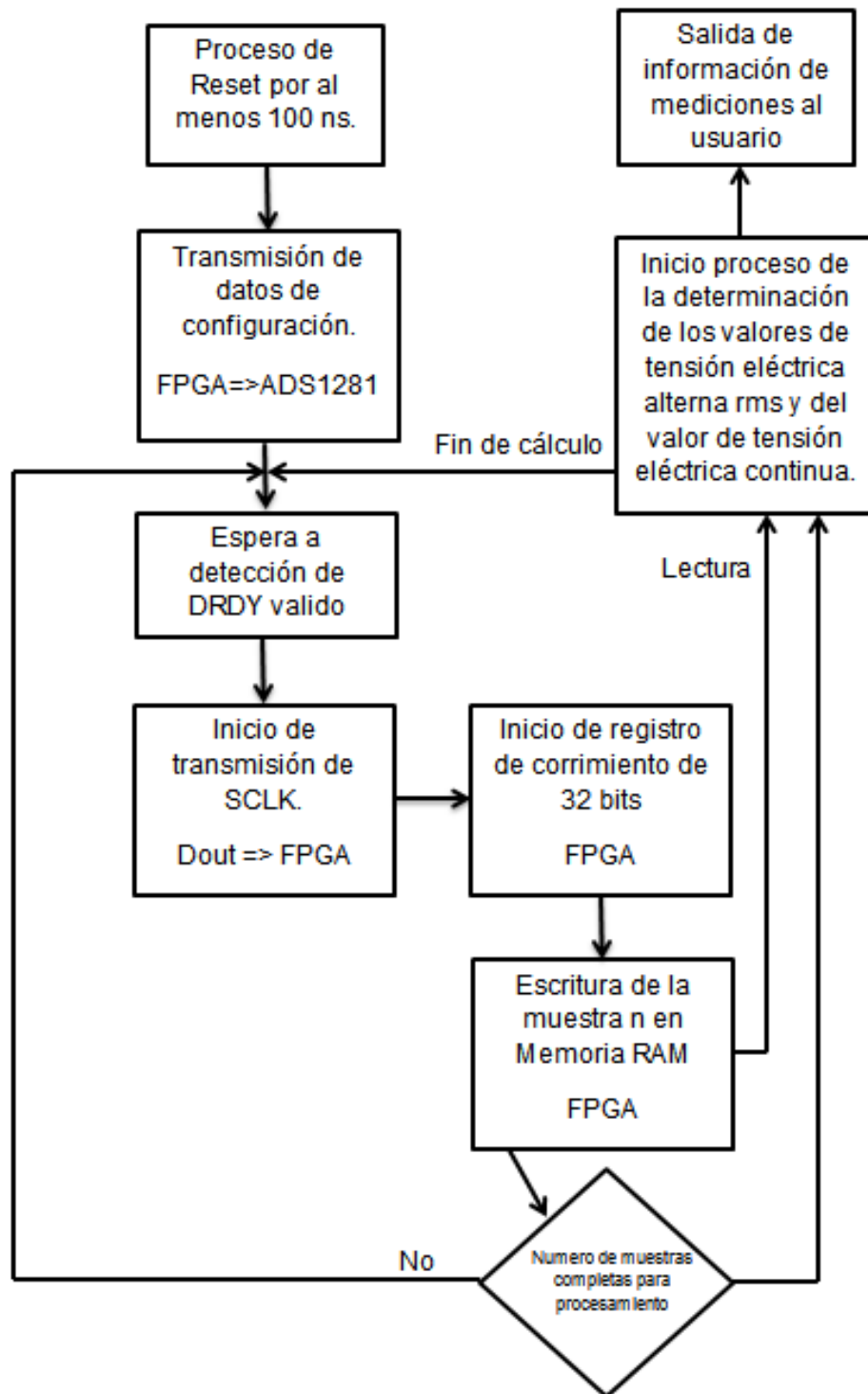


Figura 3.12. Diagrama de bloques del control del ADC por parte del FPGA.

Una vez iniciado el sistema, el primer paso es mandar un estado de reset a ambos componentes, el convertidor analógico-digital y el sistema digital implementado en el FPGA.

El siguiente paso es utilizar la interfaz SPI para configurar el convertidor analógico-digital, seguido a esto entramos a un estado de espera de la indicación del convertidor analógico-digital para dato válido, cuando esto sucede y en concordancia con la figura 3.10, se envía el SCLK a 256 kHz para obtener los bits correspondientes a las muestras. Dichas muestras son almacenadas en un módulo de $n \times 32$ de memoria RAM nativa en el FPGA, donde n es la dimensión a utilizar para la determinación de los valores de tensión eléctrica alterna y tensión eléctrica continua.

Una vez que este espacio de memoria asignado para recolección de las muestras a procesar se encuentra lleno, el siguiente estado es el cálculo del valor de tensión eléctrica continua y de tensión eléctrica alterna de la señal de interés a partir del uso de una ventana tipo Hanning, algoritmo descrito en el capítulo II de este trabajo de investigación.

El algoritmo planteado en la ecuación 8, a pesar de ser correcto, en la práctica emite resultados que producen mediciones con dispersiones mayores a las pretendidas en este trabajo de investigación. Debido a la influencia del amplificador de instrumentación completamente diferencial que se encuentra a la entrada del convertidor analógico-digital, dicha etapa fue construida con componentes de alto desempeño eléctrico, minimizando los efectos de desbalance en el amplificador. En la práctica existe aún una pequeña componente de tensión eléctrica continua que debe ser removida de la señal de interés para poder obtener un valor de tensión eléctrica alterna confiable. El algoritmo para la determinación de valores rms usando técnicas en el dominio del tiempo con ventanas es capaz de realizar esta minimización de offset con una simple modificación.

La medición de un valor de tensión eléctrica continua se puede hacer utilizando la ventana coseno seleccionada, a partir de la ecuación 12.

$$V_{DC}[f(n)] \cong \frac{Gw}{N} \sum_{n=0}^{N-1} f(n)w(n) \quad (12)$$

donde Gw es la ganancia de la ventana coseno a utilizar, para el caso de la ventana Hanning se puede comprobar que su media es 0.5, por lo que la ganancia para esta ventana es 2. Una vez que se tiene evaluado la componente de corriente continua que contiene la señal muestreada, la determinación del valor RMS de la señal puede ser calculada a partir de la ecuación 13. En la figura 4 se muestra el algoritmo para la determinación del valor RMS teniendo en cuenta una posible componente de tensión eléctrica continua.

$$V_{RMS}[f(n)] \cong \sqrt{\frac{\frac{1}{N} \sum_{n=0}^{N-1} [w(n) * (f(n) - V_{DC}[f(n)])]^2}{0.375}} \quad (13)$$

En la figura 3.13, se muestra la secuencia del algoritmo para la determinación de los valores de tensión eléctrica continua y tensión eléctrica alterna. En la figura 3.14 se muestra la arquitectura implementada en el FPGA para resolver matemáticamente el algoritmo propuesto en la ecuación 13.

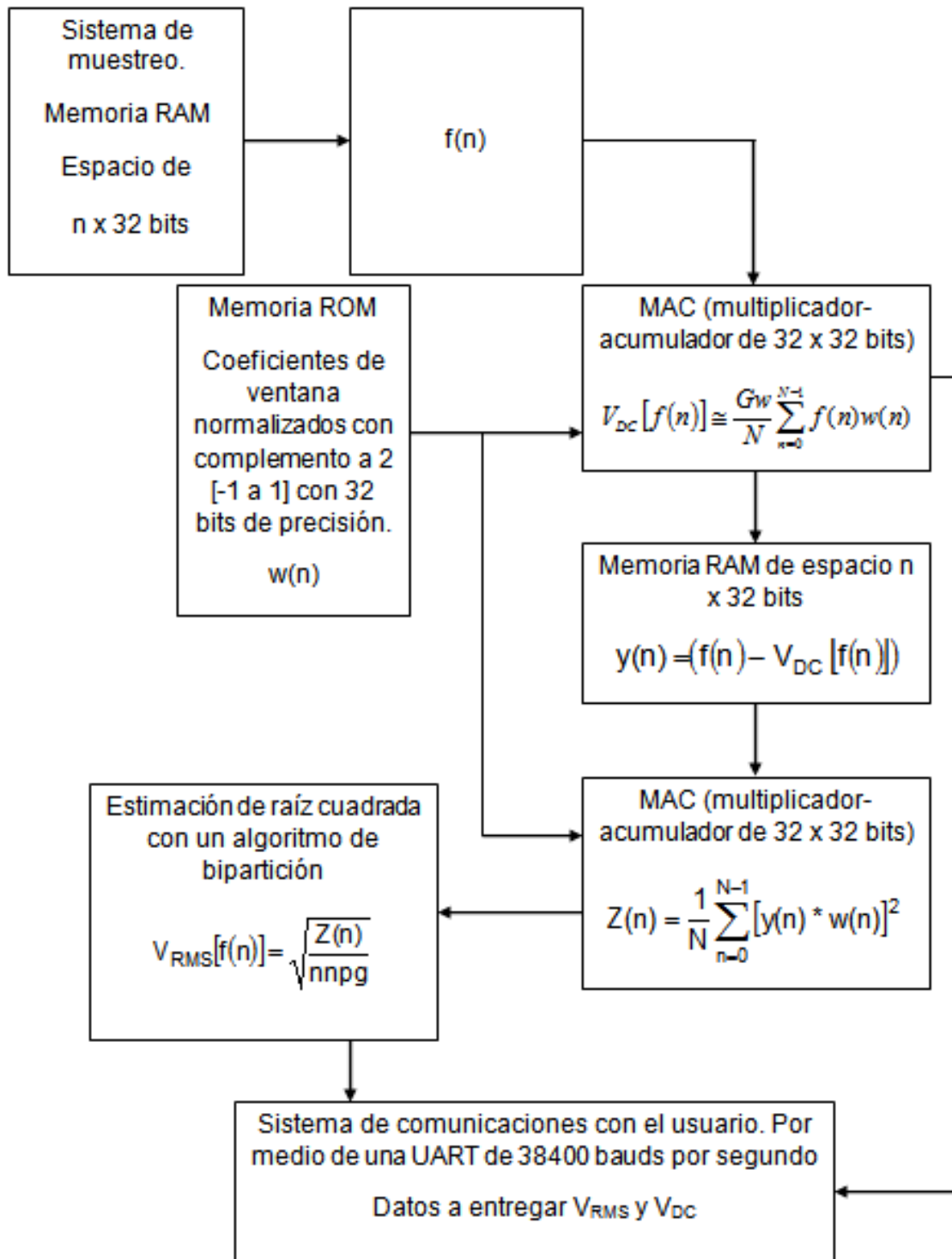


Figura 3.13. Algoritmo para la determinación del valor rms de una señal periódica de tensión eléctrica continua.

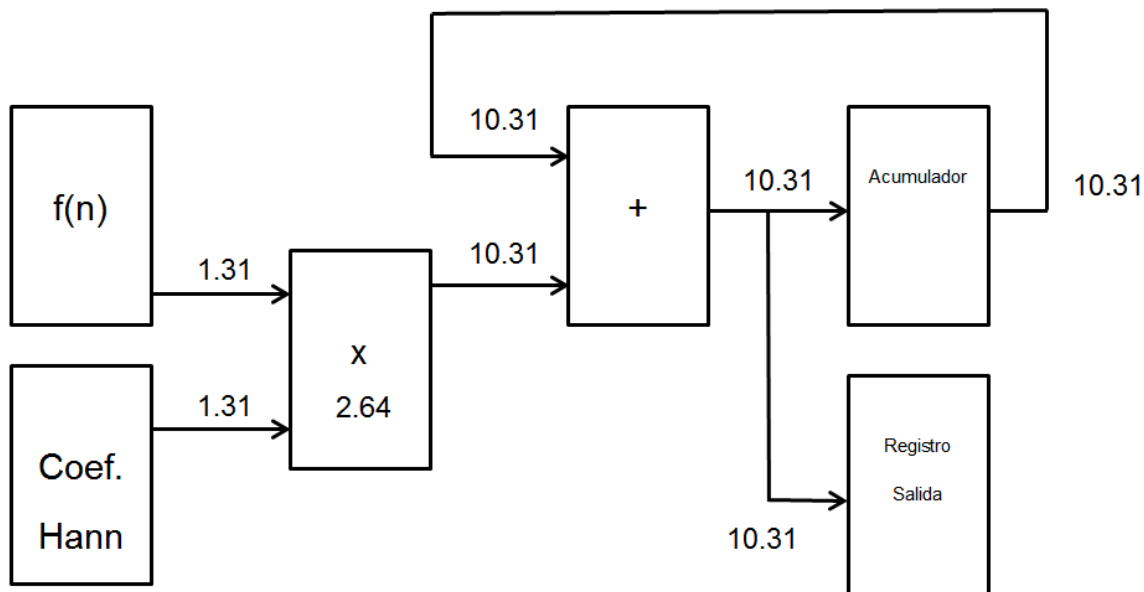


Figura 3.14. Arquitectura implementada en el FPGA para resolver matemáticamente el algoritmo; diseño de la MAC que resuelve la multiplicación de alta resolución.

La evaluación del desempeño del algoritmo para la determinación de los valores rms de tensión eléctrica alterna considera la exactitud con la que distintos tamaños de ventana influyen en el comportamiento del sistema de muestreo. Se generaron señales sintéticas de 500 mV y 700 mV a 50 Hz y 60 Hz y se probaron tamaños de venta de 64, 128, 256, 512 y 1024 puntos, se muestran los resultados de estos experimentos.

La elección del tamaño de ventana es consistente con un error sistemático despreciable en la contribución de desviaciones y ruido del sistema de muestreo. Se presentan los resultados de esta evaluación.

Adicionalmente se comprueba que la ventana elegida (Hanning) es ideal para la determinación de valores rms de señales periódicas de tensión eléctrica alterna a partir de muestreo no síncrono, esto significa que el muestreo de la señal de interés no tiene que comenzar por cruce por cero, incrementando de manera

significativamente el desempeño en el tiempo, es decir, la relación tiempo de procesamiento contra repetibilidad de las mediciones es la ideal para ventanas de 1024 puntos en comparación con sistemas comerciales de alta exactitud. Se muestran los resultados de este proceso de caracterización.

3.11 Diseño de experimentos para la evaluación de los parámetros de desempeño del convertidor analógico – digital Δ - Σ .

Las principales características a evaluar del convertidor analógico-digital son:

1. Dispersión de las mediciones a corto plazo, para mediciones de tensión eléctrica continua y de tensión eléctrica alterna. Este proceso de mediciones se realizó a partir de mediciones con método directo utilizando un dispositivo Zener Fluke 732B para continua y un calibrador multifunciones 5720A caracterizado contra un patrón de transferencia térmica Fluke 792A.
2. Evaluación de linealidad del sistema de muestreo utilizando una curva conocida de un calibrador multifunciones Fluke 5720A. En el intervalo de medición de 100 mV a 700 mV a 60 Hz.
3. Evaluación del sistema de muestreo en un caso de aplicación para mediciones en conjunto con un divisor resistivo de 15 V a 120 V a 60 Hz.

4. RESULTADOS.

En el presente capítulo se presentan los resultados de la caracterización de cada etapa que compone el sistema digital de muestreo basado en un convertidor analógico-digital Δ - Σ ADS1281. El capítulo está dividido en 3 secciones clave:

1. Evaluación y caracterización del hardware diseñado e implementado para el sistema de muestreo de alta resolución.
2. Evaluación de la aportación sistemática del algoritmo para la determinación del valor rms de una señal de tensión eléctrica alterna de baja frecuencia. Este algoritmo está basado en la técnica de ventana, se presentan los resultados de la contribución por aritmética finita en el sistema embebido en el FPGA.
3. Evaluación de las cualidades metrológicas del sistema de muestreo como un sistema de medición completo comparado contra sistemas de medición de referencia de tensión eléctrica continua a 60 Hz.

Al final del capítulo se presenta una discusión sobre los resultados obtenidos, las posibles aplicaciones dentro del campo de metrología eléctrica de alta exactitud, específicamente en el área de corriente continua y baja frecuencia. Así mismo se emiten recomendaciones sobre las posibles aplicaciones futuras de la tecnología diseñada e implementada.

4.1 Evaluación del hardware diseñado para el sistema de muestreo digital basado en un ADC Δ - Σ ADS1281.

La primera aportación relevante de este trabajo de investigación es el diseño de la tarjeta de circuito impreso, el cual tiene como principal cualidad la separación de los retornos de corriente a la fuente, estos se diseñaron de manera

separada. El convertidor analógico digital requiere de las siguientes tensiones para operar de manera correcta. Adicionalmente tiene la cualidad de minimizar los problemas de compatibilidad electromagnética y de efectos indeseables de baja frecuencia.

1. ± 12 V que sirven como alimentación para la etapa del divisor resistivo que proporciona las tensiones de referencia y de alimentación del convertidor analógico digital de alta calidad para el convertidor analógico-digital.
2. ± 5 V que sirven como alimentación a una etapa de regulación que proporciona ± 2.5 V de alimentación para el amplificador de instrumentación completamente diferencial. De esta misma tensión se regulan 3.3 V que proporcionan el nivel de tensión digital necesario para la operación del convertidor analógico-digital y de los aisladores digitales seleccionados.

En la figura 4.1 se muestra el diseño de la tarjeta de circuito impreso desarrollado para el convertidor analógico-digital.

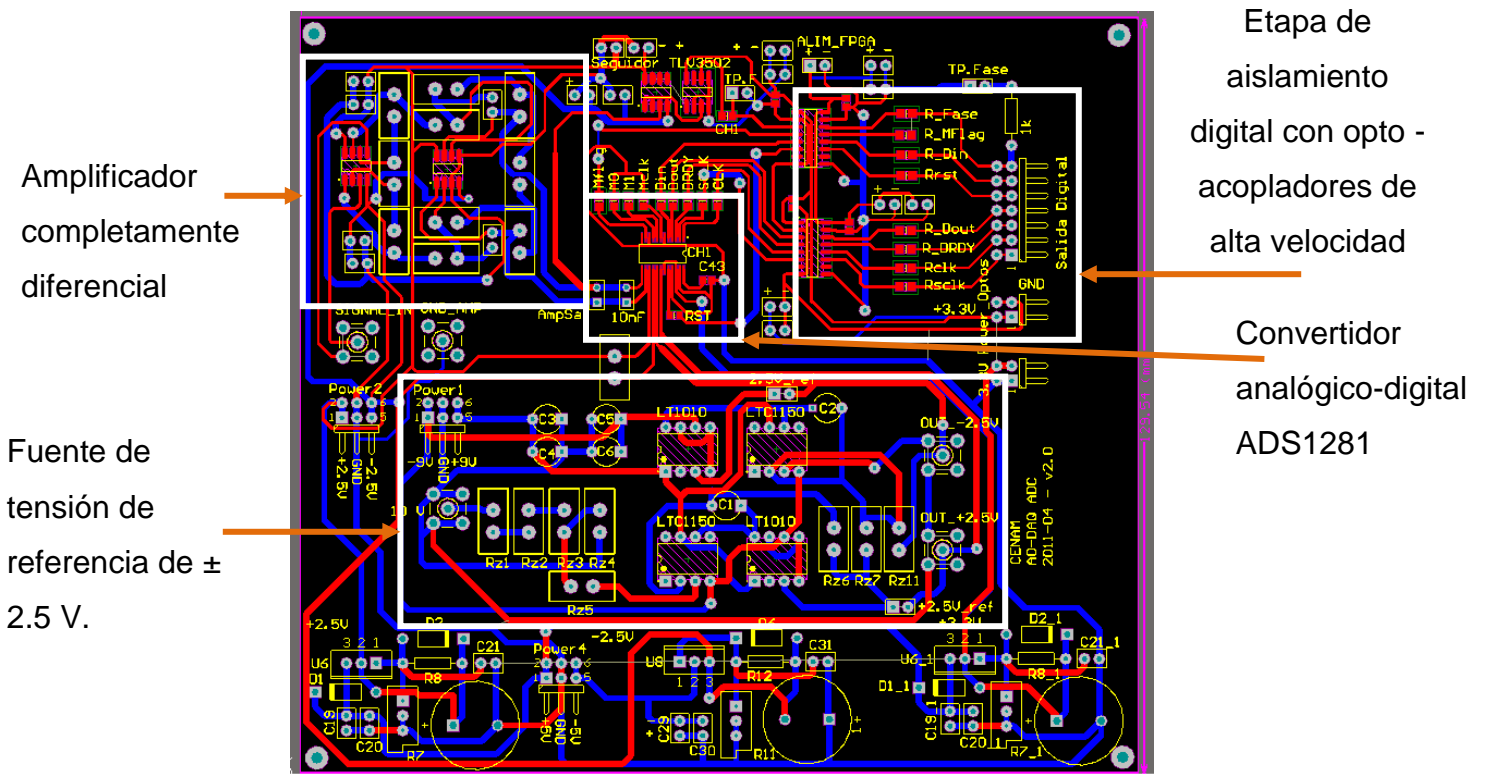


Figura 4.1. Diseño implementado de tarjeta de circuito impreso para el convertidor analógico-digital.

En la figura 4.2 se aprecia una fotografía del montaje físico del sistema de muestreo digital utilizando el diseño mostrado en la figura 4.1 en conjunto con la tarjeta de desarrollo Xilinx Spartan 3A-DSP 1800A.

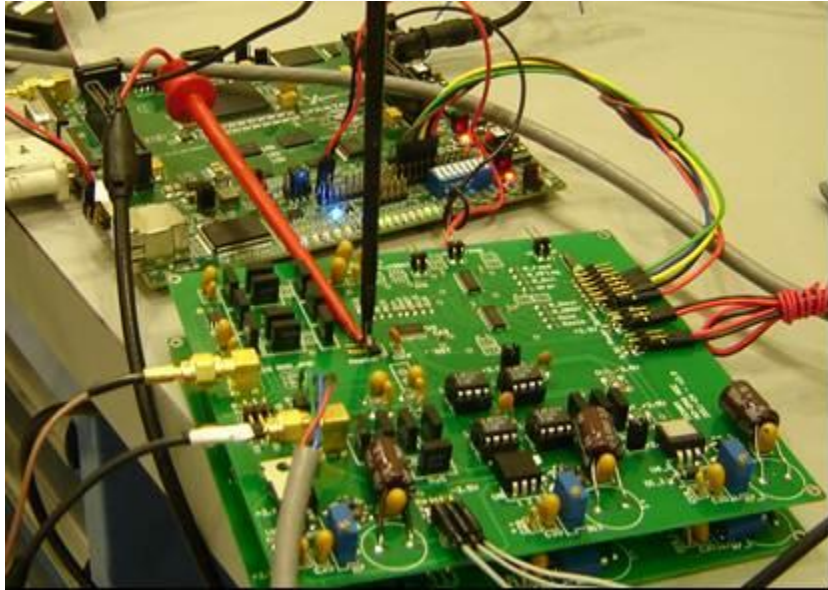


Figura 4.2. Sistema de muestreo basado en un convertidor analógico-digital Δ - Σ y un sistema FPGA Xilinx Spartan 3A-DSP 1800A.

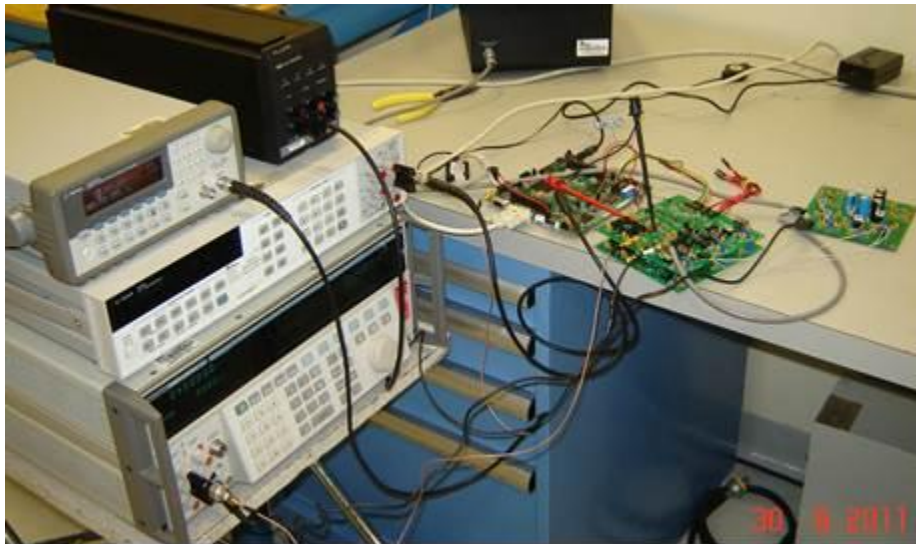


Figura 4.3. Fotografía tomada durante el proceso de caracterización del sistema de muestreo digital.

En la figura 4.3 se aprecia el conjunto de instrumentos de alta exactitud requeridos para la caracterización del sistema de muestreo propuesto. Este

conjunto de instrumentos patrón son el Calibrador Multifunciones Fluke 5720A, el Multímetro digital de 8 ½ dígitos, el generador de onda arbitraria Agilen 33250A, el dispositivo Zener 732B.

4.2 Resultados de la caracterización de la fuente de referencia a partir de un Zener de 10 V Fluke 732B.

La figura 4.4 muestra una fracción de todo el proceso de caracterización de la fuente de referencia de ± 2.5 V. El valor de referencia tiene un valor medio de 2.499 994 V, con una estabilidad a corto plazo (dentro del periodo de 18 a 24 horas) de $0.2 \mu\text{V/V}$ con mediciones cada 10 segundos, el resultado de estabilidad de la etapa negativa de la fuente de referencia tiene un comportamiento similar. La estabilidad de la fuente de referencia de tensión eléctrica continua durante días es menor a $0.5 \mu\text{V/V}$. Es importante que el ruido de la fuente se mantenga por debajo de $0.6 \mu\text{V/V}$, el cual es valor declarado para no-linealidad integral por el fabricante.

El objetivo de la caracterización de las señales de referencia, es asegurar que los valores V_{RMS} sean trazables a valores de patrones nacionales de tensión eléctrica, el cual es reproduce la unidad del volt a través del efecto Josephson Cuantico, este es mantenido en el Centro Nacional de Metrologia. Adicionalmente se requiere que los valores V_{RMS} sean trazables a los valores de los patrones nacionales de diferencia ca-cc que mantienen la unidad del volt en tensión eléctrica alterna.

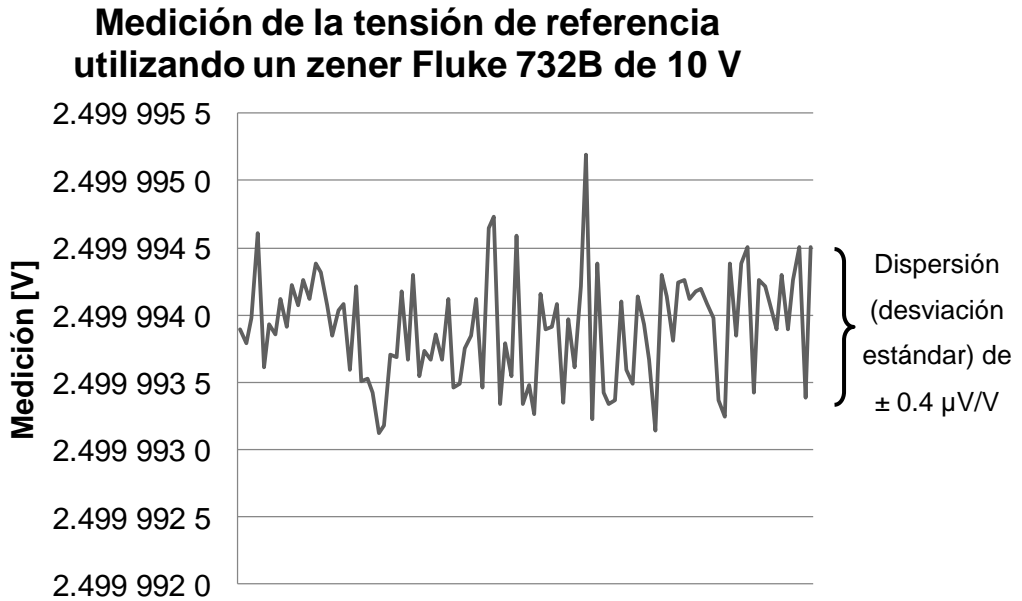


Figura 4.4. Proceso de caracterización de la fuente de referencia de 2.5 V.

4.3 Resultados de la caracterización de ruido del convertidor analógico-digital ADS1281.

Se utilizó análisis espectral para determinar la amplitud del ruido asociado al convertidor analógico-digital y del sistema de muestreo. Para el análisis se utilizó una ventana hanning similar a la utilizada en el algoritmo para la determinación del valor rms.

La tensión de prueba fue de 700 mV rms a 50 Hz, dicha tensión fue obtenida del calibrador multifunciones de alta exactitud. En la figura 4.5 se muestran los resultados.

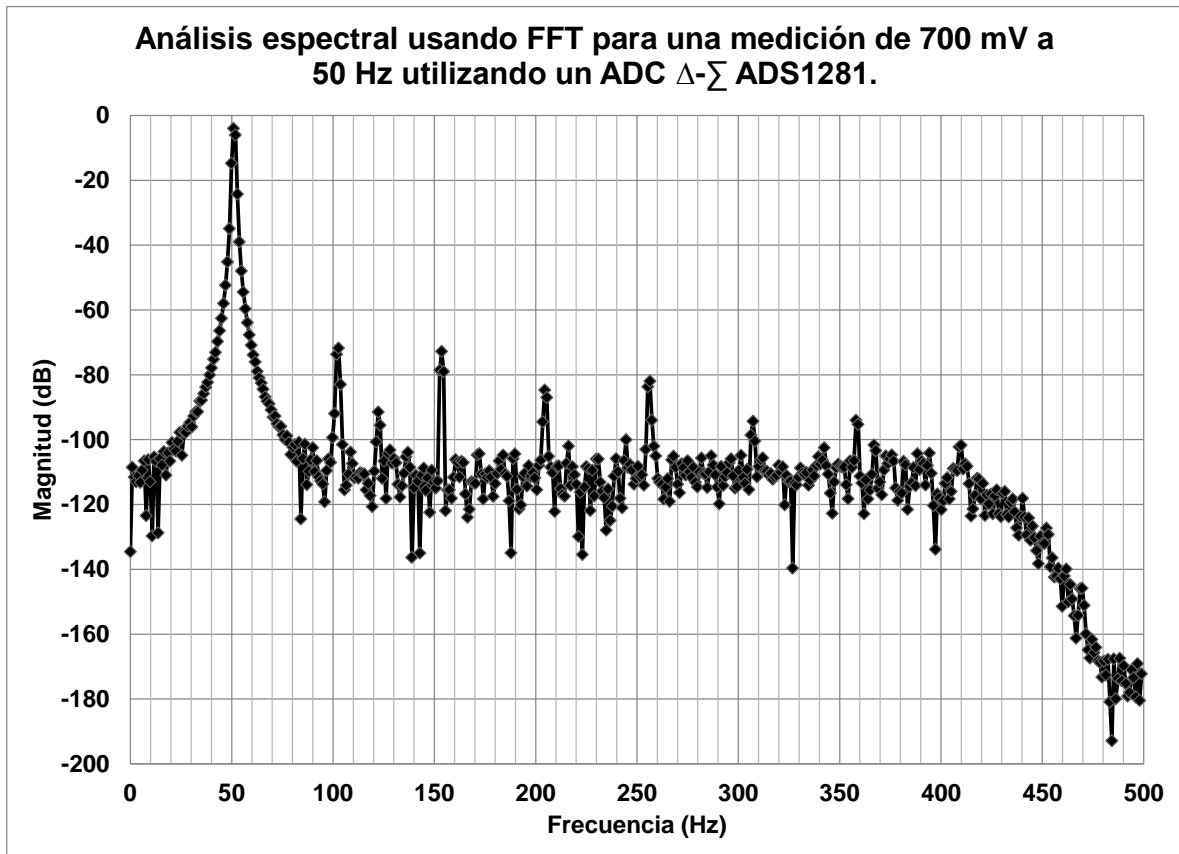


Figura 4.5 Análisis espectral utilizando la transformada rápida de Fourier y la ventana Hanning de 1024 puntos.

Los resultados muestran un ruido de fondo de -120 dB, lo que corresponde a 21 bits efectivos para el nivel de tensión muestreado, esto demuestra que el diseño de la tarjeta de circuito impreso se corresponde con la máxima capacidad del convertidor comercial, limitada únicamente por cuestiones físicas del convertidor analógico-digital.

4.4 Resultados de la caracterización de los aisladores digitales.

La finalidad de separar el mundo digital del mundo analógico del prototipo de sistema de muestreo es que los retornos de tierra que están conectados al convertidor analógico-digital fueran inmunes al ruido producido por el sistema digital de alta velocidad. Los resultados se muestran en la figura 4.6.

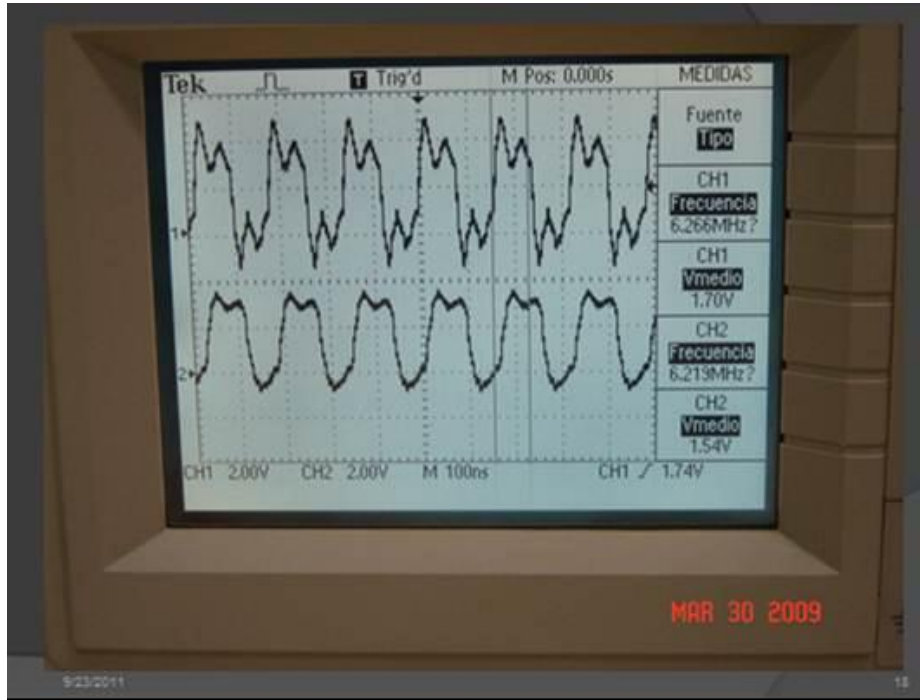


Figura 4.6 Formas de onda medidas en la entrada del circuito aislador H.P. HCPL-091J.

En la figura 4.6 se aprecian dos formas de onda, la primera señal corresponde a una señal de prueba de amplitud 3.3 V a 6.2 MHz proporcionada por la tarjeta de desarrollo Spartan 3A-DSP 1800A, la conexión fue hecha al común o tierra de la tarjeta de desarrollo. La segunda señal corresponde a la salida del circuito de aislamiento digital HCPL-091J, se aprecia una disminución de ruido pico de alrededor de 0.5 V, así como una reducción en los cambios bruscos en la señal durante los flancos de subida y de bajada. El máximo retardo medible fue de 4 ns, dando un ancho de banda de 250 MHz, el circuito aislador podría ser utilizado a su máxima capacidad confiando en la que el retraso máximo es menor a 20 ns. Las señales con frecuencia de mayor valor presentes en el prototipo es son de 4.096 MHz, por lo que se demostró que las señales de frecuencia no están comprometidas ni en amplitud, ni en tiempo.

4.5 Resultados de la caracterización del amplificador de instrumentación completamente diferencial.

El objetivo de la caracterización del amplificador de instrumentación completamente diferencial es demostrar que tiene una respuesta plana en frecuencia, caracterizar la estabilidad de la ganancia del amplificador, un alto rechazo de modo común de entrada además de mostrar que entrega la suficiente potencia para manejar el ADC, sin perder las cualidades de medición de alta exactitud.

La caracterización del amplificador de entrada del sistema digital, se da en base a tres factores, ruido, el cual es medido a partir de la desviación estándar de las mediciones, linealidad del amplificador completamente diferencial, respuesta en frecuencia.

Para esta caracterización se utilizó un multímetro 3458A en conjunto con el algoritmo Swerlein (1991), y se da en dos partes, primero se aplicó tensión eléctrica alterna de 300 mV a 700 mV a 50 Hz al amplificador y a la salida de este se midió con el algoritmo Swerlein. En un segundo paso se midió el calibrador multifunciones Fluke 5720A directamente con el 3458A y el Swerlein, los resultados de esta caracterización se pueden observar en la figura 4.7 y en la tabla 4.1.

Comportamiento del amplificador totalmente diferencial en amplitud contra un sistema de referencia basado en calibrador multifunciones de alta exactitud calibrador contra el algoritmo Swerlein en los puntos de prueba a frecuencia de 60 Hz.

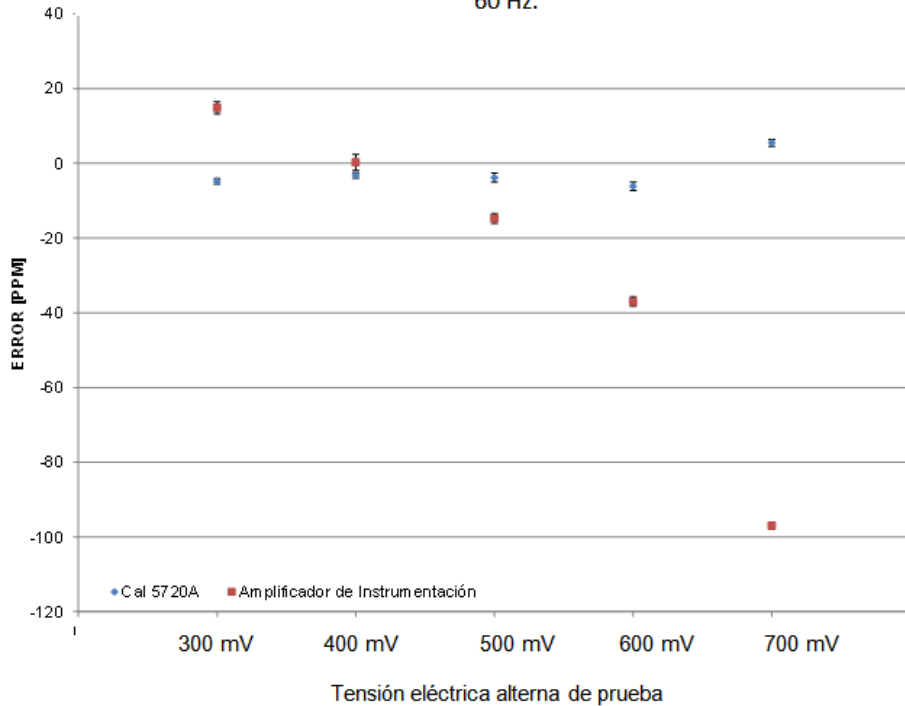


Figura 4.7 Resultado de la caracterización del amplificador de instrumentación completamente diferencial utilizando el algoritmo Swerlein.

Tabla 4.1 Resultados de la caracterización de ruido del amplificador de instrumentación completamente diferencial utilizando el algoritmo Swerlein.

Unidades relativas dadas en $\mu V/V$					
Frecuencia 60 Hz					
Amplitud	300 mV	400 mV	500 mV	600 mV	700 mV
Ruido del Amplificador de instrumentación completamente diferencial.	1.3	1.3	0.8	1.0	1.3
Ruido del calibrador multifunciones 5720A.	1.2	1.1	0.9	0.8	0.9
Contribución de ruido del amplificador de instrumentación.	-0.1	-0.2	0.1	-0.2	-0.4

De la tabla 4.1 se puede deducir que el ruido del amplificador se encuentra en el mismo nivel que el ruido del calibrador multifunciones Fluke 5720A, con una incertidumbre máxima de $\pm 0.4 \mu\text{V/V}$. Estos resultados aseguran que el comportamiento del amplificador no enmascara el comportamiento del ADC $\Delta-\Sigma$. Para la caracterización de linealidad del amplificador de instrumentación, el análisis se hace con base en el valor de ganancia para cada una de las amplitudes suministradas. Se puede notar que si se aplica un ajuste lineal, y este permanece en el tiempo, es decir estable, los errores por no linealidades van hasta los $\pm 20 \mu\text{V/V}$ y si por el contrario se utiliza una interpolación polinomial de 5to. Orden las no linealidades serán menores a $\pm 10 \mu\text{V/V}$. Esto se aprecia en la figura 4.8 y tabla 4.2.

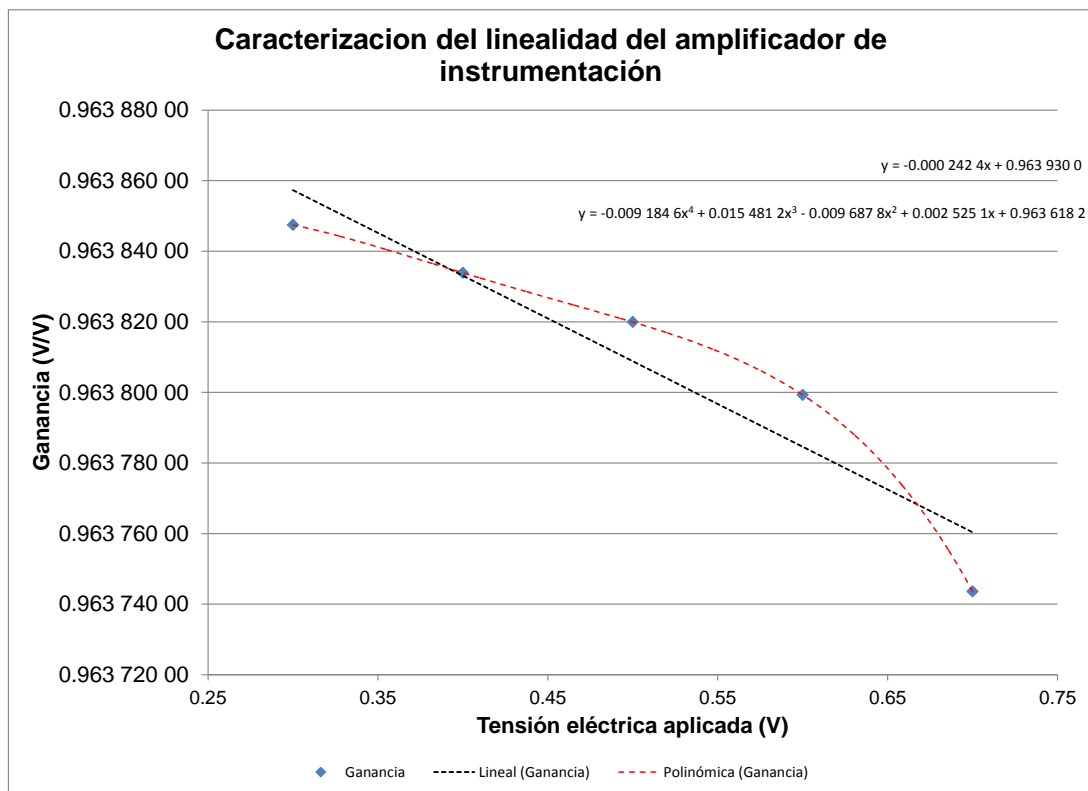


Figura 4.8. Resultado de la caracterización del amplificador de instrumentación completamente diferencial utilizando el algoritmo Swerlein para linealidad utilizando la ganancia del amplificador como parámetro.

Tabla 4.2 Diferencias máximas por linealidad en amplificador completamente diferencial.

Tendencia polinomial	0.963 847	0.963 834	0.963 820	0.963 802	0.963 751
Tendencia lineal	0.963 857	0.963 833	0.963 808	0.963 784	0.963 760
Medición	0.963 847	0.963 833	0.963 819	0.963 799	0.963 743
Error por ajuste lineal ($\mu\text{V/V}$)	-10	1	11	15	-18
Error por ajuste polinomial ($\mu\text{V/V}$)	0	0	-1	-3	-9

En lo que respecta a la frecuencia, el amplificador completamente diferencial tiene un comportamiento lineal. El coeficiente asociado es del orden de $0.1 \text{ } [\mu\text{V/V}]/\text{Hz}$, es decir 0.1 partes por millón por cada Hz. La figura 4.9 muestra el resultado de este efecto, en mediciones de 0.7 V en frecuencias de 20 Hz a 100 Hz , usando como referencia un multímetro de $8 \frac{1}{2}$ dígitos en forma de muestreo en conjunto con el algoritmo Swerlein.

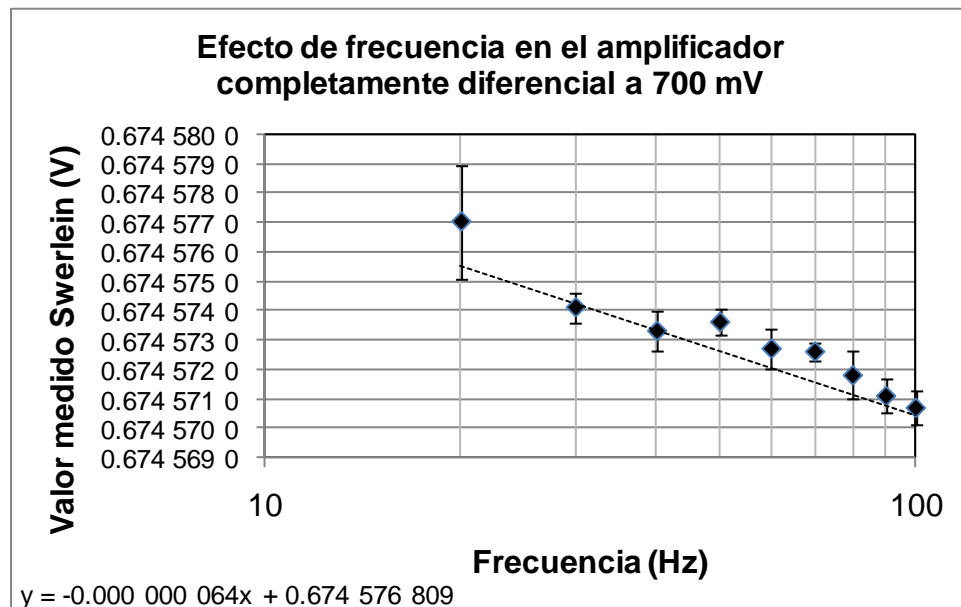


Figura 4.9 Efecto de frecuencia en el amplificador completamente diferencial.

4.6 Resultados de la caracterización del algoritmo para la determinación del valor rms de una señal periódica de baja frecuencia.

La implementación se realizó sobre un FPGA Xilinx-XC3SD1800A. El algoritmo fue probado con señales sintetizadas de amplitud variable para valores RMS de 0.5 V y de 0.7 V, y frecuencias de 50 Hz y 60 Hz. La frecuencia de muestreo de 1 kHz con un orden de 128, 256, 512 y 1024 puntos para la ventana Hanning. La amplitud de la palabra digital con la que se probó es de 24 bits. Para estas señales se simuló el error de cuantización, tal como lo que produciría un convertidor analógico-digital de 24 bits a la señal digitalizada.

Los principales componentes de esta arquitectura están dados por una ROM de coeficientes, donde se respaldan los valores de la ventana coseno a utilizar. Esta forma de aplicar los coeficientes resulta en un sistema robusto ya que se puede implementar ventanas distintas para evaluar su desempeño para la determinación de valores RMS.

El segundo punto a considerar en el diseño de la arquitectura son dos MAC (unidades de multiplicadores-acumuladores binarios), dichas unidades tienen una exactitud que ronda en el orden de los 32 bits, se tuvo especial cuidado en el diseño para que acepte números binarios en complemento a dos de manera nativa.

El tercer punto esencial en el diseño de la arquitectura es la implementación del cálculo de la raíz cuadrada por medio de un algoritmo de bipartición en un registro de aproximaciones sucesivas. Este cálculo tiene una resolución de 36 bits. Adicionalmente se requiere hacer uso de los bloques de memoria RAM con los que cuenta el FPGA seleccionado, donde la sincronía de los procesos es fundamental para obtener un sistema confiable y robusto.

En la tabla 4.3, se podrá apreciar los resultados obtenidos a partir de señales sintéticas, las cuales fueron creadas para poder evaluar las desviaciones

en la determinación del valor RMS para distintos órdenes utilizando la ventana Hanning. Se muestra que el error sistemático por el uso de esta ventana coseno depende directamente del número de muestras por ciclos así como del número de ciclos completos. Para mediciones eléctricas alternas se requiere un algoritmo que pueda tener desviaciones menores a $1 \mu\text{V/V}$, por lo que es posible lograrlo con ventanas de orden superior a 256 muestras, entregando resultados similares para 50 Hz y 60 Hz en condiciones de ventanas de 1024 puntos.

Tabla 4.3 Resultados de la caracterización del algoritmo para la determinación de valores RMS utilizando ventana Hanning.

Valor RMS simulado 500 mV			Valor RMS simulado 700 mV		
Orden de la Ventana N	Desviaciones del valor RMS determinado con el algoritmo de ventana con respecto a un valor de referencia nominal. [$\mu\text{V/V}$]		Orden de la Ventana N	Desviaciones del valor RMS determinado con el algoritmo de ventana con respecto a un valor de referencia nominal. [$\mu\text{V/V}$]	
	50 Hz	60 Hz		50 Hz	60 Hz
64	-54.914 5	-20.913 8	64	-54.896 6	-20.921 1
128	-1.047 8	-0.830 4	128	-1.029 9	-0.815 6
256	-0.145 2	-0.109 1	256	-0.127 3	-0.085 4
512	-0.085 1	-0.091 2	512	-0.067 2	-0.065 2
1024	-0.084 6	-0.089 3	1024	-0.066 7	-0.066 3

4.7 Resultados de la caracterización del sistema de muestreo digital comparado contra sistemas de referencia para medición de tensión eléctrica alterna y tensión eléctrica continua con alta exactitud.

Esta sección presenta los resultados de la caracterización del convertidor Δ - Σ para determinar el desempeño en cuanto a:

- Desempeño para resolución efectiva.
- Desempeño como medidor de tensión eléctrica alterna, en comparación con dos de los mejores medidores de tensión eléctrica alterna disponibles en el mercado.

Las mediciones fueron hechas a 1.018 V en corriente continua y de 64 mV a 517 mV a 60 Hz en corriente alterna.

La medición de 1.018 V fue llevada a cabo utilizando un dispositivo Zener Fluke 732B, el cual es caracterizado directamente contra el patrón nacional del volt, el efecto Josephson. Este procedimiento permite alcanzar incertidumbre por debajo de $\pm 0.9 \mu\text{V/V}$.

Las mediciones de tensión eléctrica alterna fueron hechas utilizando un calibrador multifunciones Fluke 5720A, el cual fue calibrado contra una patrón de transferencia térmica Fluke 792A, el cual es un patrón de transferencia AC/DC, el cual es trazable a valores de patrones nacionales mantenidos en Cenam.

Tomando como referencia el análisis de resolución efectiva mostrado por Arpaia et al (2003) y por Iuzzolino et al (2009), la resolución efectiva puede ser calculada a partir de la ecuación 14.

$$RE = \log_2 \left(\frac{\text{Escala_Completa_ADC}}{e_{RMS}} \right) \quad (14)$$

Donde la escala completa para este caso particular es ± 1.25 V, y la desviación estándar representa un nivel de ruido, e_{RMS} , en el sistema de muestreo. La tabla 4.4 muestra los resultados de la resolución efectiva para mediciones de tensión eléctrica continua y para tensión eléctrica alterna. Los resultados están en el orden de 20 a 22 bits.

Tabla 4.4 Resultados de la caracterización de la resolución efectiva del sistema de muestreo basado en un convertidor analógico-digital Δ - Σ .

Tensión de prueba [V]	Frec. [Hz]	Incertidumbre tipo A [μ V/V] Tiempo de promediación 1 s	RE [bits]	Incertidumbre tipo A [μ V/V] Tiempo de promediación 10 s.	RE [bits]
1.018	0	1.4	21	0.4	21
0.064	60	9.1	21	2.6	22
0.129	60	6.9	20	2.2	22
0.258	60	2.1	21	0.5	23
0.388	60	1.7	20	0.6	22
0.517	60	1.3	20	0.5	22

La figura 4.10 muestra los resultados de las mediciones de tensión eléctrica continua sin procesamiento adicional. La figura 4.11 muestra las mediciones para el punto de 517 mV a 60 Hz, utilizando una ventana de 1024 puntos tipo Hanning sin procesamiento adicional.

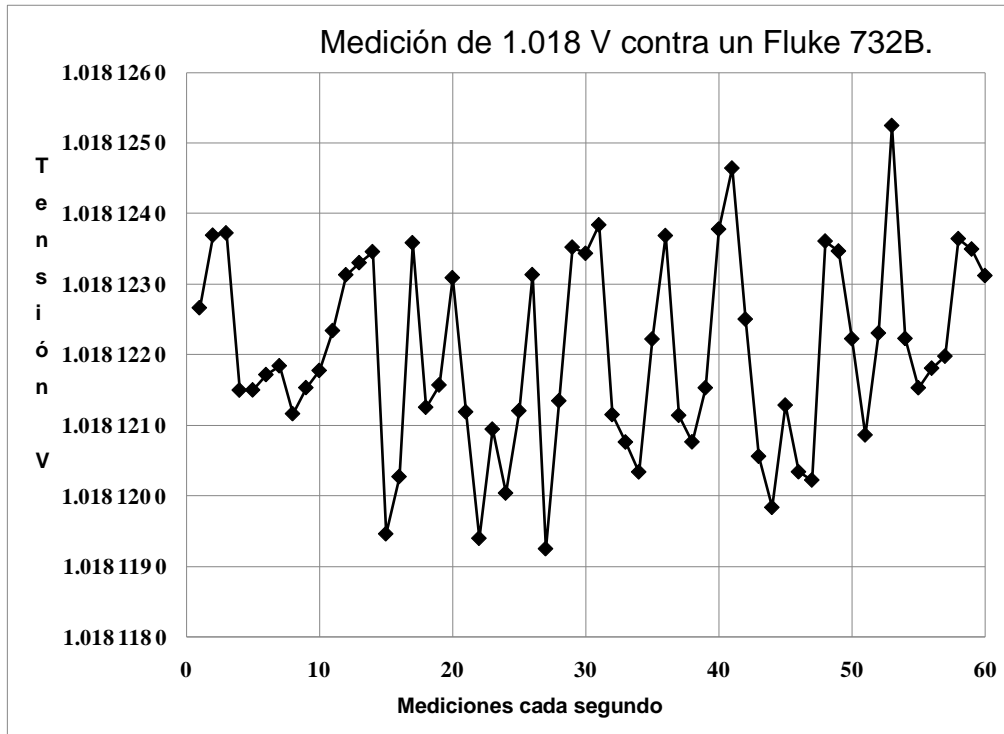


Figura 4.10 Medición de tensión eléctrica continua a 1.018 V.

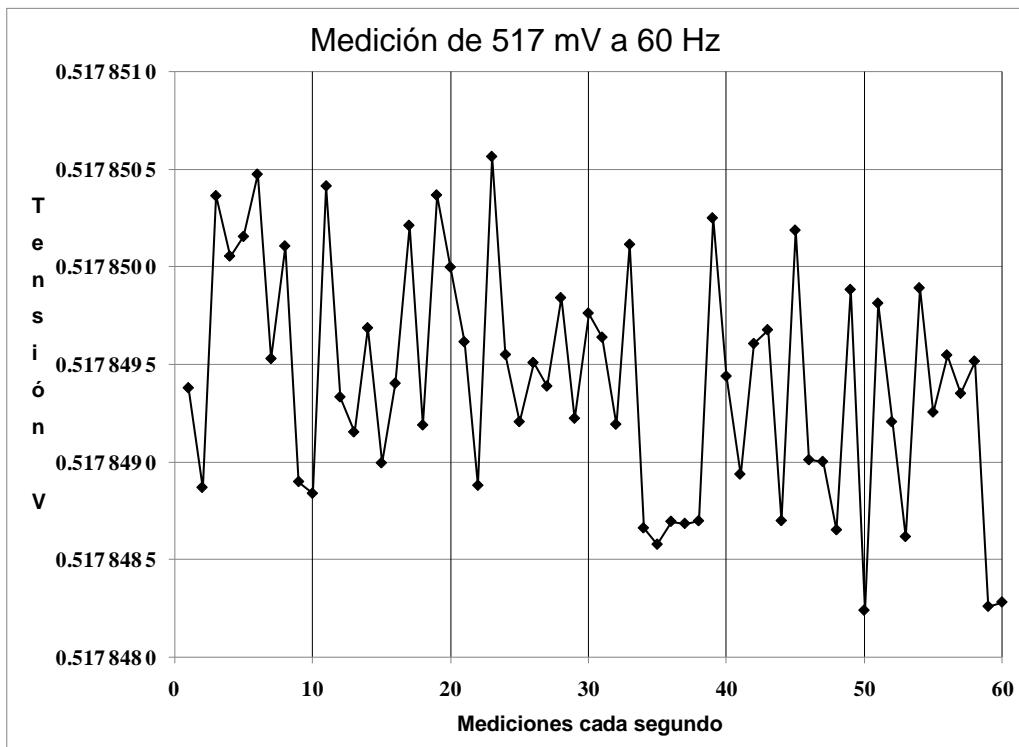


Figura 4.11 Medición de tensión eléctrica alterna a 517 mV a 60 Hz, muestra el desempeño con una desviación estándar menor a ± 695 nV.

La figura 4.12 muestra la medición del sistema de muestreo con errores relativos menores a $\pm 5 \mu\text{V/V}$; la relación de no linealidad integral se encuentra en ese intervalo también.

Tensiones eléctricas continua de 64 mV hasta 517 mV a 60 Hz están relacionadas directamente con tensiones de 15 V a 120 V a 60 Hz, para esta atenuación fue implementado un divisor resistivo con los resistores Vishay z201, con impedancia de entrada por encima de 1 M Ω .

Utilizando tiempos de promediación mayores a 1 segundo, la incertidumbre tipo A es mejorada en un factor de 5 veces, esta mejora se traduce en incremento de la resolución efectiva de hasta 1 bit o -3 dB.

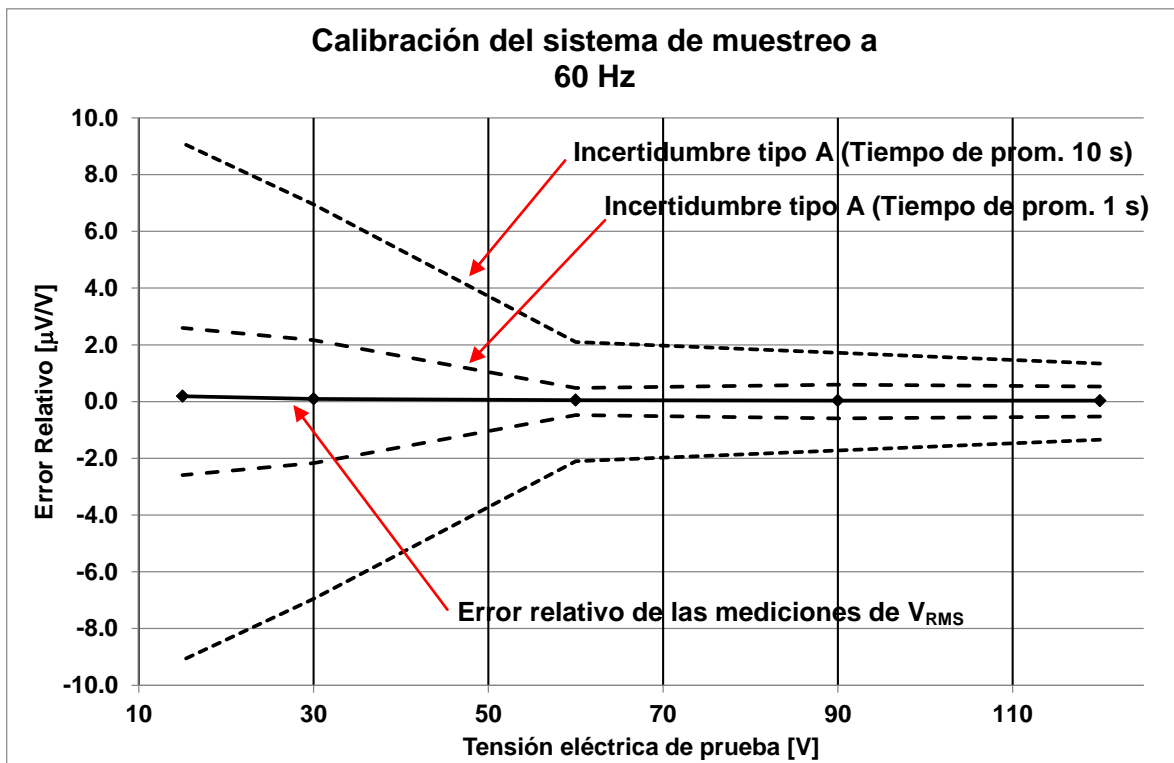


Figura 4.12 Medición desde 15 V a 120 V a 60 Hz, muestra incertidumbre tipo A para dos tiempos de promediación distintos.

La comparación del sistema de muestreo desarrollado contra medidores de tensión eléctrica alterna disponibles en el mercado fue llevada a cabo. Se utilizó un multímetro de 8 ½ dígitos en dos modos distintos, medición de tensión alterna en modo normal (default) y medición utilizando algoritmo Swerlein, adicionalmente se midió un Fluke 5790A el cual es un medidor de tensión eléctrica alterna basado en transferencias térmicas embebido en un instrumento digital. Los resultados experimentales están en la tabla 4.5. En la figura 4.13 se muestran de manera gráfica estos resultados.

Tabla 4.5 Resultados experimentales de la comparación de instrumentación comercial disponible contra el sistema de muestreo desarrollado.

Tensión de prueba [V] Frec. 60Hz	Incertidumbre tipo A [μ V/V] Modo Normal HP 3458A.	Incertidumbre tipo A [μ V/V] Algoritmo Swerlein HP 3458A.	Incertidumbre tipo A [μ V/V] Tiempo 1 s	Incertidumbre tipo A [μ V/V] Tiempo 10 s	Incertidumbre tipo A [μ V/V] Fluke 5790A.
120	6.7	0.2	1.3	0.5	2.7
90	8.9	0.7	1.7	0.6	3.3
60	14.2	0.2	2.1	0.5	2.4
30	15.0	1.2	6.9	2.2	3.1
15	17.9	0.8	9.1	2.6	2.9

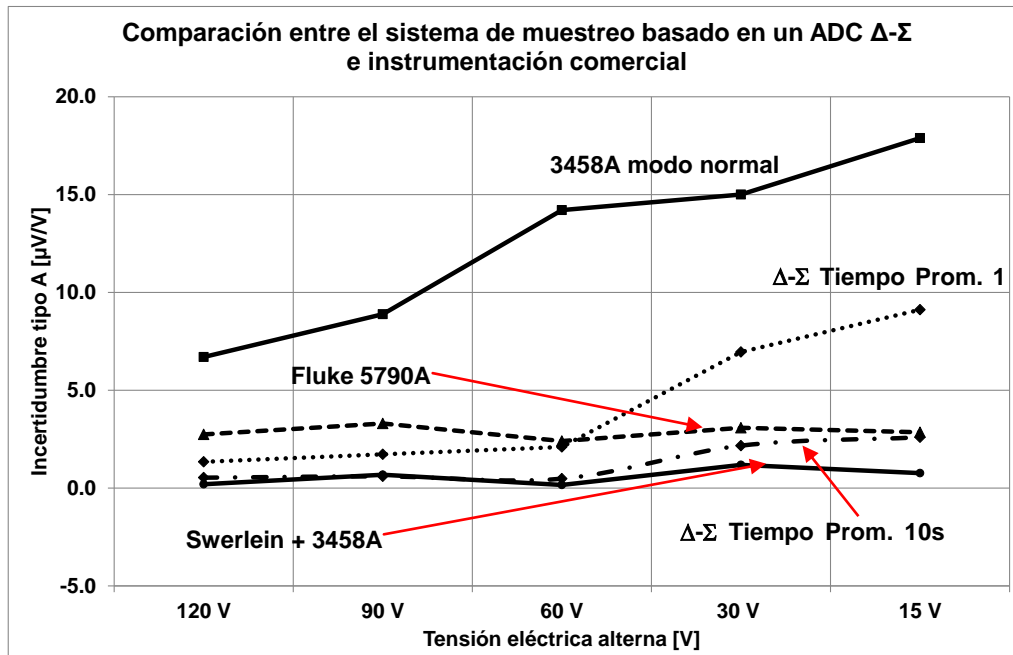


Figura 4.13 Calibración desde 15 V a 120 V a 60 Hz, compara 2 instrumentos comerciales contra el sistema de muestreo desarrollado.

El H.P. 3458A es uno de los mejores medidores de tensión eléctrica alterna en el mundo, principalmente usado por su alta exactitud, alta estabilidad, alta resolución, siendo el sistema de muestreo desarrollado comparable en desempeño en algunos puntos especialmente en 60 Hz.

El instrumento basado en el ADC Δ - Σ tiene algunas ventajas sobre la instrumentación comercial en mediciones de tensión eléctrica continua a 60 Hz: como lo son menor tiempo de procesamiento, menor incertidumbre tipo A, menor costo y portabilidad. La figura 4.14 resalta el punto referido al tiempo de procesamiento contra incertidumbre tipo A obtenida.

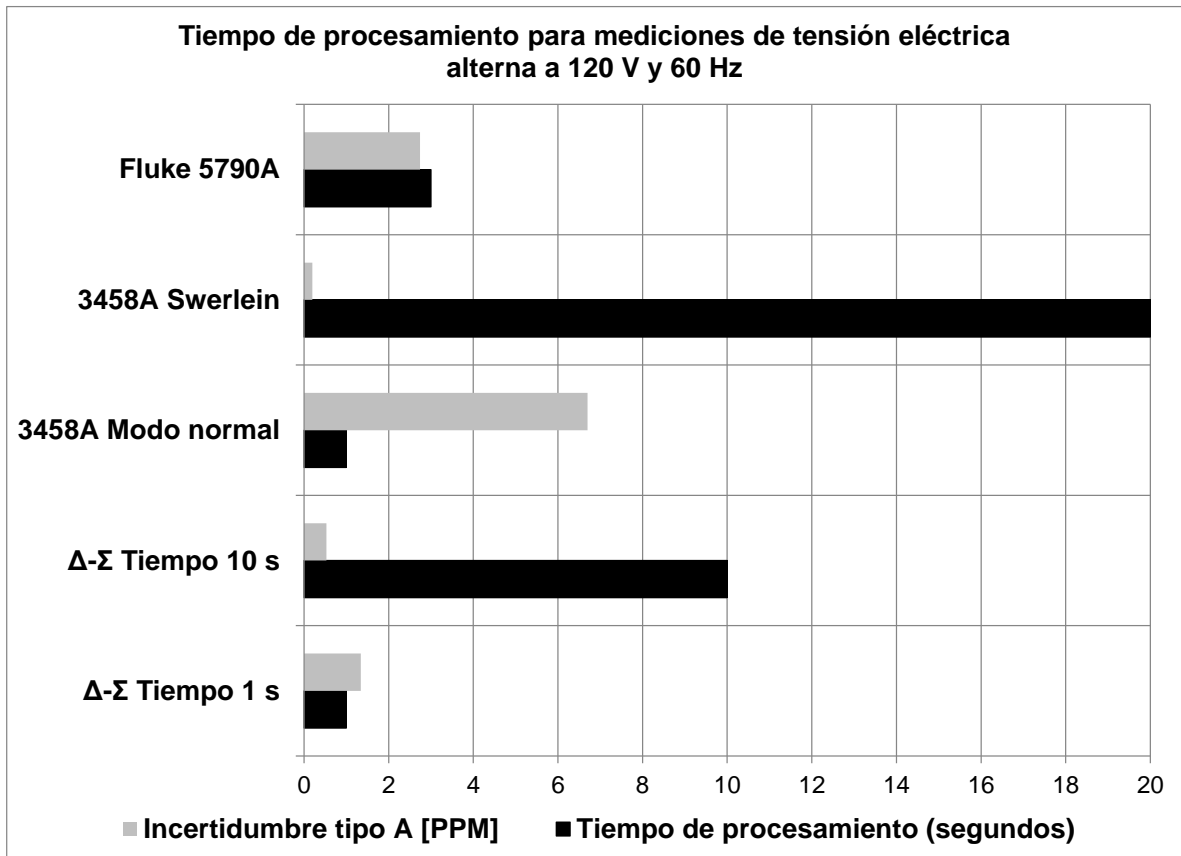


Figura 4.14 Comparación del tiempo de procesamiento contra la incertidumbre tipo A obtenidos en una medición de tensión eléctrica alterna a 120 V y 60 Hz.

5. DISCUSIÓN DE RESULTADOS.

Haciendo referencia a la hipótesis inicial y al objetivo general del tema de investigación, se puede aseverar que con los resultados obtenidos por el sistema de muestreo propuesto se cumplen ambos a cabalidad.

Se desarrolló e implemento un sistema de medición portable para tensión eléctrica continua y tensión eléctrica alterna, la evaluación de la evidencia objetiva del desempeño de este sistema basado en un convertidor analógico-digital Δ - Σ comercial, demuestra que el sistema como tal es un candidato a convertirse en un potencial voltmetro de alta exactitud para el área de metrología eléctrica en baja frecuencia. Aplicaciones en mediciones de tensión eléctrica continua de bajo nivel, en mediciones de resistencia eléctrica, en corriente eléctrica continua en niveles de 100 μ A a 1 A, en mediciones de temperatura de contacto por ejemplo usando termopares o sensores de resistencia de platino, existen además aplicaciones de tensión eléctrica alterna donde el sistema de muestreo es ideal, por ejemplo en mediciones de potencia eléctrica activa, potencia eléctrica reactiva, energía, tensión eléctrica alterna, corriente eléctrica alterna.

Esta plataforma puede ser usada en sistemas de medición de referencia a 60 Hz, como por ejemplo mediciones de corriente eléctrica alterna en combinación con derivadores de corriente, en mediciones de tensiones por encima de 120 V en combinación con divisores resistivos o divisores inductivos compensados, en sistemas de potencia y energía utilizando configuraciones multi-nucleo con más de un convertidor analógico-digital sincronizados, entre otras.

La metodología desarrollada en base al diseño, construcción y caracterización de un sistema de muestreo basado en un convertidor analógico-digital resulto ser altamente efectiva. Permitió comparar el sistema de muestreo contra sistemas de referencia en tensión eléctrica continua y en tensión eléctrica alterna. No fue necesario incluir pruebas estandarizadas y que están mayormente

dirigidas a los fabricantes de los ADC, para poder determinar el comportamiento de los principales parámetros de desempeño. Se presentó una metodología clara que sirve para evaluar ADCs que prometan alto desempeño.

Algunas características del prototipo desarrollado basado en un ADC Δ - Σ , que lo identifican como una buena opción respecto de medidores de alta exactitud de tipo comercial, son una mínima dispersión estándar obtenida en tiempo menor de medición en comparación con otros medidores comerciales y su portabilidad. En comparación con Swerlein más un H.P. 3458A, este necesita apoyo de un procesador matemático externo al 3458A, lo que resulta en tiempos de procesamiento muy grandes.

El algoritmo Swerlein utiliza ventanas rectangulares y detección de cruce por cero de la señal de interés. Esta etapa de detección incrementa la incertidumbre del método, debido al corrimiento de este cruce durante el muestreo. Si bien se ha sugerido un compensación por el tiempo de apertura del convertidor SAR del H.P. 3458A, este requiere de post-procesamiento, haciendo muy difícil la aplicación de mediciones de alta exactitud en tiempo real.

El sistema propuesto en este trabajo de investigación, es capaz de determinar valores de tensión eléctrica continua y de tensión eléctrica alterna, en tiempo real, con un procesamiento en un tiempo menor a 1 ms. Este avance se obtiene al estar embebido el procesador matemático en un dispositivo de lógica reconfigurable. Siendo esta la principal diferencia con otros métodos para mediciones de valor RMS que hay en el mercado.

Como trabajo a futuro queda mejoras en el Hardware con la finalidad de obtener mediciones aún más exactas. Una completa caracterización utilizando patrones de mayor nivel como lo son directamente contra el efecto Josephson, el cual materializa la unidad del volt en México.

6. CONCLUSIONES.

Del presente trabajo de investigación se pueden concluir los siguientes puntos:

Se caracterizó un prototipo de muestreo en base a un ADC Δ - Σ de alta resolución en los siguientes parámetros:

- Tasa de muestreo óptima (1000 muestras por segundo).
- Resolución efectiva de 22 bits sin procesamiento adicional, utilizando un filtro de promedio móvil de 10 puntos se alcanzan resoluciones de 22 bits o bien, relación señal a ruido del orden de -122 dB.
- Nivel de ruido del orden de 850 nV, lo cual significa 0.9 μ V/V en 1 Vdc y 1.3 μ V/V en 700 mV en frecuencia de línea.
- Se diseñó, fabricó y se probó un prototipo de medición que es apto para mediciones a nivel de metrología eléctrica de baja frecuencia (DC a frecuencia de línea). Se mostró evidencia objetiva de que los criterios de diseño adoptados fueron adecuados y óptimos para el propósito de la investigación.
- Se implementó un algoritmo de procesamiento digital de señales sobre FPGA basado en ventanas de coseno elevado para la determinación de valores RMS con alta exactitud el cual tiene una aportación despreciable en cuanto la propagación de errores sistemáticos, además de múltiples beneficios en comparación con las contrapartes comerciales como lo son: bajo costo, portabilidad, diseño modular que permite el crecimiento en una arquitectura en paralelo, exactitud de partes de en 10^6 utilizando un método de muestreo no síncrono para la determinación de valores RMS de señales monótono sinusoidales, procesamiento en tiempo real de muestras e implementación de un prototipo de dimensiones reducidas.

- Se demostró que el uso de ventanas tipo coseno son útiles para la medición de señales alternas de alta pureza, la implementación sobre un FPGA permite la aplicación de este algoritmo en metrología eléctrica de baja frecuencia, abriendo un espectro amplio de posibles aplicaciones en para medición de tensión en frecuencia de línea.
- Se demostró que ventanas de orden superior a 256 puntos tienen desviaciones sistemáticas menores a $2 \times 10^{-7} V$ relativos a niveles de 100 mV a 700 Mv en implementaciones en FPGA.
- Se desarrolló una interfaz para el resguardo de registros de medición en la PC.
- En la Tabla 4.4 se observa que el desempeño del prototipo se degrada a niveles menores del 40 % del intervalo dinámico, es decir por debajo de 300 mV en la señal de entrada. Al menos tres factores tienen influencia en este comportamiento: el divisor resistivo, el amplificador totalmente diferencial y el ADC-1281 (no-linealidad integral igual a 0.6 ppm). Resultados experimentales muestran la siguiente relación:
 - Influencia divisor resistivo $< 1 \mu V/V$ desde 10 % al 100 %.
 - Influencia amplificador totalmente diferencial. Total $< 1 \mu V/V$ desde 10 % al 100% (de 15 V a 120 V a 60 Hz)
 - Influencia no-linealidad integral ADC1281 $< 1 \mu V/V$ para el intervalo de medición 40 % al 100 %.
- Una resolución efectiva de 21 bits fue observada en experimentos a 1.018 V en tensión eléctrica continua y para mediciones en alterna desde 64 mV hasta 517 mV en frecuencias de 50 a 60 Hz, el uso de tiempos de integración mayores a 1.024 segundos, mejora la resolución a 22 bits.

7. BIBLIOGRAFÍA

ADS1281, Data Sheet, High-Resolution Analog-to-Digital Converter, 130 dB, 1MSPS, 0,6 ppm INL, Dallas, Texas, Texas Instruments.

Arpaia P., Cennamo F., Daponte P., Shummy H., "Modeling and Characterization of Sigma-Delta Analog-to-Digital Converter"s, *IEEE Trans. Instrum. Meas.*, vol. 52, No. 3, June 2003.

Baker B., "How to get 23bits of effective resolution from your 24-Bit converter", *Burr-Brown*, Tucson, Arizona, September 1997.

Bryant J.M, Mixed-Signal and DSP Design Techniques, Analog Devices, páginas 3.9-3.21, 2000.

Calibration: Phylosophy in practice, Fluke Corporation, 1994, Segunda edición.

Crochiere R.E., Rabiner L.R., Interpolation and decimation of digital signals- A tutorial review, *IEEE Proceedings*, Vol. 69, No. 3, Marzo 1981, páginas 300-331.

Delta-Sigma A/D conversion technique overview, Crystal Semiconductor Data Book, 1990, páginas 12-57; 12-66.

Georgakopoulos D., Pickering J.R., Williams J.M., Wright P.S, Oversampling DELTA-SIGMA ADC for metrology applications, *IEEE Proceedings Sci Measurements technology*, Vol. 151, No 5, páginas 362-367, Septiembre 2004.

Hegedus H., Mostarac P., Malaric R., Precision RMS value measurement of non-coherent sampled signals, 2010, Conference on Precision Electromagnetic Measurements Digest CPEM 2010, 205-206.

HP 3458A Multimeter, Operating, Programming, and Configuration Manual, pp A2-A3, available at www.agilent.com.

IEEE Std 1241-2010, IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters, IEEE Instrumentation & Measurements Society.

Iuzzolino R., Palafox L., Kürten Ihlenfeld W.G., Mohns E., and Brendel C., "Design and Characterization of a Sampling System Based on Σ - Δ Analog-to-Digital

Converters for Electrical Metrology”, *IEEE Trans. Instrum. Meas.*, vol. 58, No. 4, April 2009.

Kourhy J, Tao H., Data Converters for communication systems, *IEEE Communications magazine*, Octubre 1988, páginas 113-117.

Kürten Ihlenfeld W. G., Mohns E., Behr R., Williams J.M., Patel P., Ramm G., Bachmair H., “Characterization of a high-resolution analog-to-digital converter with a Josephson AC voltage source”, *IEEE Trans. Instrum. Meas.*, vol. 54, no. 2, pp. 649–652, April. 2005.

Kürten Ihlenfeld W. G., Iuzzolino R., Otto R., Schröder K., Putensen U., and Huettemann H., “A new sampling power standard of highest accuracy for ac quantities”, in *CPEM Conf. Dig.*, 2008, pp. 208–209.

Kürten Ihlenfeld W. G. y Mohns E., Ac-dc transfer measurements of highest accuracy with synchronous analogue-to-digital conversion, 2004, *Metrologia* 41, 111-115.

Novotny M., Sedlacek M., “RMS value measurement based on classical and modified digital signal processing algorithms”, *Measurement*, 2008, ScienceDirect, vol. 41, pp. 236-250.

Palafox L., Ramm G., Behr R., Kürten Ihlenfeld W.G., Moser H., “Primary AC Power Standard Based on Programmable Josephson Junction Arrays”, *IEEE Trans. Instrum. Meas.*, vol. 56, No. 23, April 2007.

Pogliano U., Trinchera B., Francone F., Configurable unit for precise RMS measurements, 2008, Conference on Precision Electromagnetic Measurements Digest CPEM 2008, 236-237.

Romero-Troncoso, R. J., Espinosa-Flores-Verdad, G., “Phase accumulator synthesis algorithm for DDS applications”, *Electronics Letters, The Institution of Electrical Engineers*, vol. 35 no. 10, pp 770-772. ISSN 0013-5194, 1999.

Salomon O.M., “The use of DFT windows on SNR and harmonic distortion computations”, *IEEE Trans. Instrum. Meas.*, vol. 43, 1994, pp. 194–199.

Sangil Park, Principles of Sigma-Delta Modulation for Analog-to-Digital Converters, Motorola Signal Processors.

Schreier R. and Temes G. C., “Understanding Delta–Sigma Data Converters. Piscataway”, NJ: IEEE Press, 2005.

Smith S., Digital Signal Processing: “The Scientist and Engineer’s Guide”, Second Edition, California Technical Publishing, San Diego, CA, USA, pp. 277-282, 1999.

Spinelli E.M., García P.A., and Guaraglia D.O., "A Dual-Mode Conditioning Circuit for Differential Analog-to-Digital Converters", *IEEE Trans. Instrum. Meas.*, vol 59, no. 1, pp. 195-199, Jan. 2010.

Swerlein R., "A 10 ppm Accurate Digital AC Measurement Algorithm", Hewlett-Packard Co., 1991, available at <http://www.home.agilent.com>.

Tóth E., Ribeiro A.M., Debatin R.M., "Power and Energy Reference System Applying Dual-Channel Sampling", *IEEE Trans. Instrum. Meas.*, Vol 54, No 1, February 2005.

Zhang Liu Yue Xiuzeng Zhao Honggang Zou Benxia, Studies on AC voltage measurement by sampling method, 2008, Conference on Precision Electromagnetic Measurements Digest CPEM 2008, 388-390.

ANEXO I. Código VHDL para las principales etapas de procesamiento.

Código para la configuración del ADC ADS1281.

```
-----  
-- Company: CENAM  
-- Engineer: Marco Antonio Rodríguez Guerrero  
  
-- Design Name: SPI1281_CTRL  
-- Module Name: SPI_TX.vhd  
-- Project Name: SPI1281_CTRL  
-- Target Devices: XC3SD1800A-5FG676  
-- Tool versions: ISE Project Navigator 11.4  
-- Description: Comunicación SPI serial para la configuración de un ADC ADS1281  
--  
-- Dependencies:  
--  
-- Revision:  
-- Revision 1.0 - Funcional  
-- Additional Comments:  
--  
-----
```

```
library IEEE;  
use IEEE.std_logic_1164.all;  
use IEEE.std_logic_arith.all;  
use IEEE.std_logic_unsigned.all;  
  
entity SPI_TX is  
    port(  
        CLK : in std_logic;  
        RST : in std_logic;  
        Hspitx: in std_logic; --Habilita el modulo de transmision  
        Din : out std_logic;  
        SerialCLK: out std_logic;  
        Zeot : out std_logic  
    );  
end entity SPI_TX;
```

Architecture Modulos of SPI_TX is

```
component Mux_salida  
    port(  
        Sel: in std_logic_vector(5 downto 0);  
        R1: in std_logic_vector(7 downto 0);  
        R2: in std_logic_vector(7 downto 0);
```

```

        R3: in std_logic_vector(7 downto 0);
        R4: in std_logic_vector(7 downto 0);
        R5: in std_logic_vector(7 downto 0);
        Tx: out std_logic
    );
end component;

component Control
    port(
        RST : in std_logic;
        CLK : in std_logic;
        T512: in std_logic;
        Z   : in std_logic;
        Z8  : in std_logic;
        Zst : in std_logic;
        Sel : out std_logic_vector (5 downto 0);--Señal de control para el Mux de salida
        H   : out std_logic;  --Señal de habilitación para el contador de 24 ciclos de 4 MHz
        H8  : out std_logic;  --Señal de habilitación para el contador de 8 ciclos - 512 kHz
        Hst : out std_logic;
        Zeot: out std_logic
    );
end component;

component sclk is
    port (
        RST : in std_logic;
        CLK : in std_logic;
        Hsclk : in std_logic;
        Tbase : out std_logic;
        SCLK : out std_logic
    );
end component;

component Base_4MHz
    generic(
        n : integer := 4
    );
    port (
        RST : in std_logic;
        CLK : in std_logic;
        T : out std_logic
    );
end component;

```

```

component Contador_24Ciclos
  port(
    RST: in std_logic;
    CLK: in std_logic;
    H  : in std_logic;
    Z  : out std_logic
  );
end component;

component Contador_8Ciclos_512kHz
  port(
    RST: in std_logic;
    CLK: in std_logic;
    H8  : in std_logic;
    Z8  : out std_logic
  );
end component;

component Contador_start
  port(
    RST: in std_logic;
    CLK: in std_logic;
    Hst : in std_logic;
    Zst : out std_logic
  );
end component;

signal Z,Z8,Zst,H,H8,Hst,T,T512,SerialClock: std_logic;
signal R1,R2,R3,R4,R5: std_logic_vector (7 downto 0);
signal Sel: std_logic_vector (5 downto 0);

begin
  R1 <= "00010001";    -- Detiene el convertidor para ser configurado
  R2 <= "00100001";    -- Escribe al registro 1 de configuracion
  R3 <= "00000000";    -- Numero de registros a escribir - 1
  R4 <= "11000001";    -- Configuracion del ADC
                        ---
                        -- 250 SPS 000
                        -- 500 SPS 001
                        -- 1000 SPS 010
                        -- 2000 SPS 011
                        -- 4000 SPS 100
  R5 <= "00010000";    -- Inicia el ciclo de conversiones
  --Instancias
  Modulo_1 : Mux_salida port map (Sel,R1,R2,R3,R4,R5,Din);
  Modulo_2 : Control port map (RST,CLK,T512,Z,Z8,Zst,Sel,H,H8,Hst,Zeot);
  Modulo_3 : sclk port map (RST,CLK,Hspitx,T512,SerialCLK);

```

```

Modulo_4 : Base_4MHz port map (RST,CLK,T);
Modulo_5 : Contador_24Ciclos port map (RST,CLK,H,Z);
Modulo_6 : Contador_8Ciclos_512kHz port map (RST,CLK,H8,Z8);
Modulo_7 : Contador_start port map (RST,CLK,Hst,Zst);
end Modulos;

```

Modulo para la recepción de datos seriales de conversión del ADC ADS1281 y la estimación del valor RMS de una señal de baja frecuencia.

```

-----
-- Company: CENAM
-- Engineer: Marco Antonio Rodríguez Guerrero

-- Design Name: SPI1281_CTRL
-- Module Name: SPI_RX.vhd
-- Project Name: SPI1281_CTRL
-- Target Devices: XC3SD1800A-5FG676
-- Tool versions: ISE Project Navigator 11.4
-- Description: Comunicación SPI serial para la recepción de muestra, para guardado de
muestras en memoria y procesamiento para obtención de valores RMS y valores de DC
--
-- Dependencies:
--
-- Revision:
-- Revision 1.0 - Functional
-- Additional Comments:
--
-----

Library IEEE;
use IEEE.STD_LOGIC_1164.all;

entity SPI_RX is
    port(
        CLK : in std_logic;
        RST : in std_logic;
        ENBrx : in std_logic;
        DRDY: in std_logic;
        Dout: in std_logic;
        SerialCLK: out std_logic;
        Tx : out std_logic;
        Ctrl_reg: out std_logic;
        Qdata_test: out std_logic;
        SelLeds : in std_logic_vector (3 downto 0);
        Salida : out std_logic_vector (7 downto 0)
    );
end SPI_RX;

```


Architecture SPI_RX of SPI_RX is

```
component RegistroCorrimiento_32
  generic( n : integer := 32
    );
  port(
    RST      : in std_logic;
    CLK      : in std_logic;
    Dout     : in std_logic;
    CTRL: in std_logic; ---Control de carga.
    Q       : out std_logic_vector (n-1 downto 0);
    Qdata: out std_logic
  );
end component;
```

```
component Contador_100ns
  port(
    RST: in std_logic;
    CLK: in std_logic;
    OPC: in std_logic;
    Z100: out std_logic
  );
end component;
```

```
component sclk
  port (
    RST : in std_logic;
    CLK : in std_logic;
    Hsclk  : in std_logic;
    Tbase  : out std_logic;
    SCLK  : out std_logic
  );
end component;
```

```
component FSM_RX
  port(
    RST      : in std_logic;
    CLK      : in std_logic;
    ENB      : in std_logic;
    DRDY     : in std_logic;
    Dout     : in std_logic;
    Z100     : in std_logic;
    SCLK     : in std_logic;
    Z32      : in std_logic;
    Zae      : in std_logic;
    EOT232   : in std_logic;
```

```

CTRL    : out std_logic;
H       : out std_logic;  --Habilitacion para SCLK
OPC     : out std_logic;
OPCae   : out std_logic_vector (1 downto 0);
WE      : out std_logic;
H232    : out std_logic;
EN      : out std_logic;
EN2     : out std_logic;
WE2     : out std_logic;
sel     : out std_logic_VECTOR (1 downto 0);
seleccion: out std_logic_VECTOR (1 downto 0);
STF_Vcd : out std_logic;
EOF_Vcd : in std_logic;
WE3     : out std_logic;
EN3     : out std_logic;
LDA     : out std_logic;
LDR     : out std_logic;
STAr aiz : out std_logic;
EOAr aiz : in std_logic;
STF_var : out std_logic;
EOF_var : in std_logic
);
end component;

component Contador_32bits
port(
RST : in std_logic;
CLK : in std_logic;
CTRL: in std_logic;
Z32 : out std_logic
);
end component;

component RS_232_TX_ADS1281
port(
CLK : in std_logic;
RST : in std_logic;
Q   : in std_logic_vector (31 downto 0);
ENB : in std_logic;
Tx232: out std_logic;
EOT232: out std_logic
);
end component;

Component RAM
port(
clk : in std_logic;

```

```

        we : in std_logic;
        en  : in std_logic;
        addr : in std_logic_vector(9 downto 0);
        di   : in std_logic_vector(31 downto 0);
        do   : out std_logic_vector(31 downto 0)
    );
end component;

Component ContadorAE
    port(
        RST : in std_logic;
        CLK : in std_logic;
        OPC : in std_logic_vector (1 downto 0);
        Z   : out std_logic;
        I   : out std_logic_vector (9 downto 0)
    );
end component;

Component MultiplicadorEscalado
    port(
        X : in std_logic_vector (31 downto 0);
        A : in std_logic_vector (31 downto 0);
        M : out std_logic_vector (63 downto 0)
    );
end component;

Component ROM_HANN
    port(
        I : in std_logic_vector(9 downto 0);
        A : out std_logic_vector(31 downto 0)
    );
end component;

Component muxBuffer
port(
    X           : in std_logic_vector (31 downto 0);
    XH          : in std_logic_vector (31 downto 0);
    Z           : in std_logic_vector (31 downto 0);
    varianza    : in std_logic_vector (31 downto 0);
    SEL         : in std_logic_vector (1 downto 0);
    salidaBufer: out std_logic_vector (31 downto 0)
);
end component;

Component muxAddress
port(
    CLK           : in std_logic;

```

```

        AE          : in std_logic_vector (9 downto 0);
        Index_mediaMod : in std_logic_vector (9 downto 0);
        Index_var     : in std_logic_vector (9 downto 0);
        SELECCION    : in std_logic_vector (1 downto 0);
        ADDRESS      : out std_logic_vector (9 downto 0)
    );
end component;

```

```

Component mediaMod
    port(
        RST      : in std_logic;
        CLK      : in std_logic;
        STF      : in std_logic;
        X        : in std_logic_vector (31 downto 0);
        H        : in std_logic_vector (31 downto 0);
        Index    : out std_logic_vector (9 downto 0);
        XH_media: out std_logic_vector (31 downto 0);
        EOF      : out std_logic
    );
end component;

```

```

Component restador
    port(
        QRest : in std_logic_vector (31 downto 0);
        PRest : in std_logic_vector (31 downto 0);
        URest : out std_logic_vector(31 downto 0)
    );
end component;

```

```

Component Sumador_Z
    port(
        Q1 : in std_logic_vector (40 downto 0);
        P1 : in std_logic_vector (40 downto 0);
        U1 : out std_logic_vector(40 downto 0)
    );
end component;

```

```

Component Acumulador_Z
    port(
        RST : in std_logic;
        CLK : in std_logic;
        LDA1 : in std_logic;
        U1  : in std_logic_vector(40 downto 0);
        Q1  : out std_logic_vector(40 downto 0)
    );
end Component;

```

```

Component Resultado_Z
    port(
        RST : in std_logic;
        CLK : in std_logic;
        LDR1 : in std_logic;
        U1 : in std_logic_vector (40 downto 0);
        R1 : out std_logic_vector (40 downto 0)
    );
end Component;

Component Multiplicador41bits
    port(
        X2 : in std_logic_vector (40 downto 0);
        A2 : in std_logic_vector (40 downto 0);
        M2 : out std_logic_vector(40 downto 0)
    );
end Component;

Component SalidaLeds
    port(
        CLK : in std_logic;
        Xm : in std_logic_vector (31 downto 0);
        XH : in std_logic_vector (31 downto 0);
        Sel : in std_logic_vector (3 downto 0);
        Salida: out std_logic_vector (7 downto 0)
    );
end component;

Component Raiz
    port(
        RST : in std_logic;
        CLK : in std_logic;
        STA : in std_logic;
        X : in std_logic_vector(71 downto 0);
        EOA : out std_logic;
        R : out std_logic_vector(35 downto 0)
    );
end Component;

component varianza
    port(
        RST : in std_logic;
        CLK : in std_logic;
        STF : in std_logic;
        X : in std_logic_vector (31 downto 0);
        Index : out std_logic_vector (9 downto 0);
    );
end component;

```



```

Modulo_4 : FSM_RX port map
(RST,CLK,ENBrx,DRDY,Dout,Z100,SerialClock,Z32,Zae,EOT232,CTRL,Hsclk,OPC,OPCae,WE,H232,EN,EN2
,WE2,sel,seleccion,STF_Vdc,EOF_Vcd,WE3,EN3,LDA,LDR,STAr aiz,EOAraiz,STF_var,EOF_var);
Modulo_5 : Contador_32bits port map (RST,CLK,CTRL,Z32);
Modulo_6 : RS_232_TX_ADS1281 port map (CLK,RST,salidaBuffer,H232,Tx,EOT232);
Modulo_7 : RAM port map (CLK,WE,EN,ADDRESS,dataScaled,X);
Modulo_8 : ContadorAE port map (RST,CLK,OPCae,Zae,AE);
Modulo_9 : MultiplicadorEscalado port map (Q,"01001100110100111001100101101111",M
dataScaled <= M (61 downto 30);
-----
Modulo_10: mediaMod port map
(RST,CLK,STF_Vdc,X,HANN,Index_mediaMod,XH_media,EOF_Vcd);
Modulo_11: ROM_HANN port map (ADDRESS,HANN);
Modulo_12: restador port map (X,XH_media,SAMPLE);
-----
Modulo_13: RAM port map (CLK,WE2,EN2,ADDRESS,SAMPLE,Y);
Modulo_14: muxBuffer port map(X,Y,Z,varsalida,sel,salidaBuffer);
Modulo_15: muxAddress port map(CLK,AE,Index_mediaMod,Index_var,seleccion,ADDRESS);
-----
Modulo_17: RAM port map(CLK,WE2,EN2,ADDRESS,YH,Z);
Modulo_18: MultiplicadorEscalado port map(SAMPLE,HANN,YH_AUX);
YH <= YH_AUX(62 downto 31);
-----
Modulo_19: restador port map (Z,X"00000000",Zi_u);
-----
Modulo_21: varianza port map (RST,CLK,STF_var,Zi_u,Index_var,dosVAR,EOF_var);
VAR2 <= '0'&dosVAR(31 downto 1);
VAR_AMP <= "000000000" & VAR2;
-----
Modulo_22: SalidaLeds port map (CLK,VAR2,VAR2,SelLeds,Salida);
Modulo_23: RAM port map (CLK,WE3,EN3,ADDRESS,Vrms,varsalida);
Modulo_24: Multiplicador41bits port map(VAR_AMP,k,kVAR);
-----
inraiz <= kVAR(34 downto 0) & "00000000000000000000000000000000";
Modulo_25 : raiz port map(RST,CLK,STAr aiz,inraiz,EOAraiz,outraiz);
Vrms <= outraiz (34 downto 3); ---VALOR RMS---
SerialCLK <= SerialClock;
Ctrl_reg <= CTRL;
Qdata_test <= Qdata;

end SPI_RX;

```

Memoria RAM de 1024x32.

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

entity ram is
port (
    clk : in std_logic;
    we  : in std_logic;
    en  : in std_logic;
    addr : in std_logic_vector(9 downto 0);
    di   : in std_logic_vector(31 downto 0);
    do   : out std_logic_vector(31 downto 0)
);
end ram;

architecture Arreglo_generico of ram is
type ram_type is array (1023 downto 0) of std_logic_vector (31 downto 0);
signal RAM : ram_type;
begin
    process (clk)
    begin
        if clk'event and clk = '1' then
            if en = '1' then
                if we = '1' then
                    RAM(conv_integer(addr)) <= di;
                else
                    do <= RAM( conv_integer(addr));
                end if;
            end if;
        end if;
    end process;
end Arreglo_generico;
```


LUT de Coeficientes para ventana Hann de 1024 puntos con formato 1.31 de punto fijo.

```
- HANN WAVEFORM COEFFICIENTS
--
-- FI - CENAM, Enero 2011
--
-- Marco A. Rodriguez Guerrero
--

library IEEE;
use IEEE.std_logic_1164.all;

entity ROM_HANN is
  port(
    I : in  std_logic_vector(9 downto 0);
    A : out std_logic_vector(31 downto 0)
  );
end ROM_HANN;

architecture LUTable of ROM_HANN is
begin
  process(I)
  begin
    case I is
      -- Coefficient format 1.31
      when "0000000000" => A <= "00000000000000000000000000000000"; -- Index 0 Coefficient 0.0000000000
      when "0000000001" => A <= "0000000000000000000100111011110101"; -- Index 1 Coefficient 0.00000941236
      when "0000000010" => A <= "00000000000000000010011101111010011"; -- Index 2 Coefficient 0.00003764908
      when "0000000011" => A <= "000000000000000101100011010010111"; -- Index 3 Coefficient 0.00008470910
      when "0000000100" => A <= "00000000000001001110111100111111"; -- Index 4 Coefficient 0.00015059065
      when "0000000101" => A <= "0000000000001111011010111000100"; -- Index 5 Coefficient 0.00023529125
      when "0000000110" => A <= "00000000000010110001101000100000"; -- Index 6 Coefficient 0.00033880771
      when "0000000111" => A <= "00000000000011110001110001001010"; -- Index 7 Coefficient 0.00046113612
      when "0000001000" => A <= "00000000000100111011110000111001"; -- Index 8 Coefficient 0.00060227190
      when "0000001001" => A <= "00000000000110001111100111100001"; -- Index 9 Coefficient 0.00076220971
      when "0000001010" => A <= "00000000000111101101010100110101"; -- Index 10 Coefficient 0.00094094355
      when "0000001011" => A <= "00000000001001010100111000100111"; -- Index 11 Coefficient 0.00113846668
      when "0000001100" => A <= "00000000001011000110010010100110"; -- Index 12 Coefficient 0.00135477166
      when "0000001101" => A <= "00000000001101000001100010100010"; -- Index 13 Coefficient 0.00158985035
      when "0000001110" => A <= "00000000001111000110101000000111"; -- Index 14 Coefficient 0.00184369391
      when "0000001111" => A <= "00000000010001010101100011000000"; -- Index 15 Coefficient 0.00211629277
      when "0000010000" => A <= "00000000010011101110010010111000"; -- Index 16 Coefficient 0.00240763666
      when "0000010001" => A <= "00000000010110010000110111011000"; -- Index 17 Coefficient 0.00271771463
      when "0000010010" => A <= "00000000011000111101010000000101"; -- Index 18 Coefficient 0.00304651500
      when "0000010011" => A <= "00000000011011110011011100100110"; -- Index 19 Coefficient 0.00339402538
      when "0000010100" => A <= "00000000011101100110111000111110"; -- Index 20 Coefficient 0.00376023270
      when "0000010101" => A <= "000000000100001111101001111010000"; -- Index 21 Coefficient 0.00414512317
      when "0000010110" => A <= "000000000100101010000110100011101"; -- Index 22 Coefficient 0.00454868229
      when "0000010111" => A <= "000000000101000101110001011100011"; -- Index 23 Coefficient 0.00497089487
      when "0000011000" => A <= "000000000101100010101010100000010"; -- Index 24 Coefficient 0.00541174502
      when "0000011001" => A <= "000000000110000000110001101010101"; -- Index 25 Coefficient 0.00587121613
      when "0000011010" => A <= "00000000011010000000110110110110"; -- Index 26 Coefficient 0.00634929092
      when "0000011011" => A <= "00000000011100000010101000000001"; -- Index 27 Coefficient 0.00684595138
      when "0000011100" => A <= "000000000111100010011011000001011"; -- Index 28 Coefficient 0.00736117881
      when "0000011101" => A <= "00000001000000101011001110101100"; -- Index 29 Coefficient 0.00789495381
      when "0000011110" => A <= "00000001000101001100110010111001"; -- Index 30 Coefficient 0.00844725628
```

```

when "0000011111"=> A <= "00000001001001111000000100000100"; -- Index 31 Coefficient 0.00901806545
when "0000100000"=> A <= "00000001001110101101000001100000"; -- Index 32 Coefficient 0.00960735980
when "0000100001"=> A <= "00000001010011101011101010011101"; -- Index 33 Coefficient 0.01021511716
when "0000100010"=> A <= "0000000101100011001111110001010"; -- Index 34 Coefficient 0.01084131464
when "0000100011"=> A <= "00000001011100001011101110100"; -- Index 35 Coefficient 0.01148592867
when "0000100100"=> A <= "00000001100011100001100010100111"; -- Index 36 Coefficient 0.01214893498
when "0000100101"=> A <= "00000001101001000110110001101110"; -- Index 37 Coefficient 0.01283030861
when "0000100110"=> A <= "0000000110110110101101000010001"; -- Index 38 Coefficient 0.01353002390
when "0000100111"=> A <= "00000001110100101110000101011000"; -- Index 39 Coefficient 0.01424805451
when "0000101000"=> A <= "00000001111010110000001000001001"; -- Index 40 Coefficient 0.01498437340
when "0000101001"=> A <= "00000001000000011101101111101000"; -- Index 41 Coefficient 0.01573895286
when "0000101010"=> A <= "000000010000111010000111010111000"; -- Index 42 Coefficient 0.01651176448
when "0000101011"=> A <= "000000010001101101111101000111011"; -- Index 43 Coefficient 0.01730277915
when "0000101100"=> A <= "000000010010100010111111000110001"; -- Index 44 Coefficient 0.01811196710
when "0000101101"=> A <= "000000010011011001001101001011000"; -- Index 45 Coefficient 0.01893929787
when "0000101110"=> A <= "000000010100010000100111001101110"; -- Index 46 Coefficient 0.01978474029
when "0000101111"=> A <= "000000010101001001001101000101110"; -- Index 47 Coefficient 0.02064826255
when "0000110000"=> A <= "000000010110000010111110101010010"; -- Index 48 Coefficient 0.02152983213
when "0000110001"=> A <= "00000001011011110111101110010100"; -- Index 49 Coefficient 0.02242941585
when "0000110010"=> A <= "00000001011111010000100010101001"; -- Index 50 Coefficient 0.02334697982
when "0000110011"=> A <= "000000011000110111011000001001001"; -- Index 51 Coefficient 0.02428248952
when "0000110100"=> A <= "000000011001110101110111000100111"; -- Index 52 Coefficient 0.02523590970
when "0000110101"=> A <= "000000011010110101100000111110111"; -- Index 53 Coefficient 0.02620720449
when "0000110110"=> A <= "00000001101110110010101010101010"; -- Index 54 Coefficient 0.02719633731
when "0000110111"=> A <= "000000011100111000010101000101111"; -- Index 55 Coefficient 0.02820327092
when "0000111000"=> A <= "00000001110111101101111011110110"; -- Index 56 Coefficient 0.02922796741
when "0000111001"=> A <= "00000001111011111110011001101100"; -- Index 57 Coefficient 0.03027038820
when "0000111010"=> A <= "00000100000000101010001100111100"; -- Index 58 Coefficient 0.03133049404
when "0000111011"=> A <= "00000100001001011111010000010000"; -- Index 59 Coefficient 0.03240824503
when "0000111100"=> A <= "00000100010010011101100010010010"; -- Index 60 Coefficient 0.03350360058
when "0000111101"=> A <= "00000100011011100101000001101001"; -- Index 61 Coefficient 0.03461651946
when "0000111110"=> A <= "00000100100100110101101100111100"; -- Index 62 Coefficient 0.03574695976
when "0000111111"=> A <= "00000100101110001111100010101101"; -- Index 63 Coefficient 0.03689487893
when "0001000000"=> A <= "00000100110111110010100001100010"; -- Index 64 Coefficient 0.03806023374
when "0001000001"=> A <= "00000101000001011110100111111011"; -- Index 65 Coefficient 0.03924298033
when "0001000010"=> A <= "00000101001011010011110100011000"; -- Index 66 Coefficient 0.04044307415
when "0001000011"=> A <= "00000101010101010010000101011010"; -- Index 67 Coefficient 0.04166047004
when "0001000100"=> A <= "00000101011111011001011001011101"; -- Index 68 Coefficient 0.04289512215
when "0001000101"=> A <= "0000010110100110100110111011110"; -- Index 69 Coefficient 0.04414698400
when "0001000110"=> A <= "00000101110100000011000100011000"; -- Index 70 Coefficient 0.04541600845
when "0001000111"=> A <= "0000010111111010010101100000011"; -- Index 71 Coefficient 0.04670214774
when "0001001000"=> A <= "00000110001001010000101000011000"; -- Index 72 Coefficient 0.04800535344
when "0001001001"=> A <= "00000110010100000100110011101101"; -- Index 73 Coefficient 0.04932557648
when "0001001010"=> A <= "00000110011111000001111000011000"; -- Index 74 Coefficient 0.05066276715
when "0001001011"=> A <= "00000110101010000111110100101101"; -- Index 75 Coefficient 0.05201687512
when "0001001100"=> A <= "00000110110101010110100110111110"; -- Index 76 Coefficient 0.05338784940
when "0001001101"=> A <= "00000111000000101110001101011100"; -- Index 77 Coefficient 0.05477563838
when "0001001110"=> A <= "00000111001100001110100110010111"; -- Index 78 Coefficient 0.05618018980
when "0001001111"=> A <= "000001110101111011101111111110"; -- Index 79 Coefficient 0.05760145078
when "0001010000"=> A <= "00000111100011101001101000011101"; -- Index 80 Coefficient 0.05903936783
when "0001010001"=> A <= "00000111101111100100001110000001"; -- Index 81 Coefficient 0.06049388679
when "0001010010"=> A <= "00000111111011100111011110110011"; -- Index 82 Coefficient 0.06196495290
when "0001010011"=> A <= "00001000000111110011011000111101"; -- Index 83 Coefficient 0.06345251079
when "0001010100"=> A <= "0000100001010000011111010100111"; -- Index 84 Coefficient 0.06495650445
when "0001010101"=> A <= "00001000100000100101000001110111"; -- Index 85 Coefficient 0.06647687724
when "0001010110"=> A <= "00001000101101001010101001100110"; -- Index 86 Coefficient 0.06801357194
when "0001010111"=> A <= "00001000111001111000111001011011"; -- Index 87 Coefficient 0.06956653068
when "0001011000"=> A <= "00001001000110101111100101110110"; -- Index 88 Coefficient 0.07113569500

```

```

when "0001011001"=> A <= "00001001010011101110110000000011"; -- Index 89 Coefficient 0.07272100582
when "0001011010"=> A <= "00001001100000110110010110000010"; -- Index 90 Coefficient 0.07432240345
when "0001011011"=> A <= "00001001101110000110010101110010"; -- Index 91 Coefficient 0.07593982760
when "0001011100"=> A <= "00001001111011011110101101010000"; -- Index 92 Coefficient 0.07757321738
when "0001011101"=> A <= "0000101000100011111011010011000"; -- Index 93 Coefficient 0.07922251128
when "0001011110"=> A <= "00001010010110101000011011000100"; -- Index 94 Coefficient 0.08088764722
when "0001011111"=> A <= "00001010100100011001101101001110"; -- Index 95 Coefficient 0.08256856251
when "0001100000"=> A <= "00001010110010010011001110101110"; -- Index 96 Coefficient 0.08426519385
when "0001100001"=> A <= "00001011000000010100111101011011"; -- Index 97 Coefficient 0.08597747737
when "0001100010"=> A <= "0000101100111001111010111001010"; -- Index 98 Coefficient 0.08770534861
when "0001100011"=> A <= "00001011011100110000111001110000"; -- Index 99 Coefficient 0.08944874250
when "0001100100"=> A <= "00001011101011001011000010111111"; -- Index 100 Coefficient 0.09120759342
when "0001100101"=> A <= "00001011111001101101010000101011"; -- Index 101 Coefficient 0.09298183515
when "0001100110"=> A <= "00001100001000010111100000100010"; -- Index 102 Coefficient 0.09477140087
when "0001100111"=> A <= "00001100010111001001110000010100"; -- Index 103 Coefficient 0.09657622323
when "0001101000"=> A <= "00001100100110000011111101110000"; -- Index 104 Coefficient 0.09839623426
when "0001101001"=> A <= "00001100110101000110000110100010"; -- Index 105 Coefficient 0.10023136545
when "0001101010"=> A <= "00001101000100010000001000010110"; -- Index 106 Coefficient 0.10208154770
when "0001101011"=> A <= "00001101010011100010000000110111"; -- Index 107 Coefficient 0.10394671135
when "0001101100"=> A <= "0000110110001011101101101010101"; -- Index 108 Coefficient 0.10582678619
when "0001101101"=> A <= "00001101110010011101001100100000"; -- Index 109 Coefficient 0.10772170142
when "0001101110"=> A <= "00001110000010000110011010111000"; -- Index 110 Coefficient 0.10963138571
when "0001101111"=> A <= "0000111001000110111010110011010"; -- Index 111 Coefficient 0.11155576716
when "0001110000"=> A <= "0000111010000110111111100101010"; -- Index 112 Coefficient 0.11349477332
when "0001110001"=> A <= "00001110110001110000001011001011"; -- Index 113 Coefficient 0.11544833118
when "0001110010"=> A <= "000011110000011101111111100001"; -- Index 114 Coefficient 0.11741636719
when "0001110011"=> A <= "00001111010010000111010111001010"; -- Index 115 Coefficient 0.11939880726
when "0001110100"=> A <= "00001111100010011110001111101000"; -- Index 116 Coefficient 0.12139557675
when "0001110101"=> A <= "00001111110010111100100110011001"; -- Index 117 Coefficient 0.12340660048
when "0001110110"=> A <= "00010000000011100010011000111001"; -- Index 118 Coefficient 0.12543180274
when "0001110111"=> A <= "00010000010100001111100100100110"; -- Index 119 Coefficient 0.127471110728
when "0001111000"=> A <= "00010000100101000100000110111011"; -- Index 120 Coefficient 0.12952443732
when "0001111001"=> A <= "00010000110101111111111101010001"; -- Index 121 Coefficient 0.13159171556
when "0001111010"=> A <= "00010001000111000011000101000010"; -- Index 122 Coefficient 0.13367286416
when "0001111011"=> A <= "00010001011000001101011011100101"; -- Index 123 Coefficient 0.13576780478
when "0001111100"=> A <= "00010001101001011110111110010000"; -- Index 124 Coefficient 0.13787645852
when "0001111101"=> A <= "00010001111010110111101010011010"; -- Index 125 Coefficient 0.13999874602
when "0001111110"=> A <= "00010010001100010111011101010110"; -- Index 126 Coefficient 0.14213458736
when "0001111111"=> A <= "00010010011101111110010100011000"; -- Index 127 Coefficient 0.14428390213
when "0010000000"=> A <= "00010010101111101100001100110011"; -- Index 128 Coefficient 0.14644660941
when "0010000001"=> A <= "00010011000001100001000011110111"; -- Index 129 Coefficient 0.14862262777
when "0010000010"=> A <= "00010011010011011100110110110100"; -- Index 130 Coefficient 0.15081187530
when "0010000011"=> A <= "0001001110010101111100010111010"; -- Index 131 Coefficient 0.15301426956
when "0010000100"=> A <= "00010011110111101001000101010110"; -- Index 132 Coefficient 0.15522972763
when "0010000101"=> A <= "00010100001001111001011011010101"; -- Index 133 Coefficient 0.15745816611
when "0010000110"=> A <= "00010100011100010000100010000011"; -- Index 134 Coefficient 0.15969950110
when "0010000111"=> A <= "00010100101110101110010110101011"; -- Index 135 Coefficient 0.16195364821
when "0010001000"=> A <= "00010101000001010010110110010111"; -- Index 136 Coefficient 0.16422052258
when "0010001001"=> A <= "00010101010011111101111100011111"; -- Index 137 Coefficient 0.16650003885
when "0010001010"=> A <= "00010101100110101111101011011011"; -- Index 138 Coefficient 0.16879211120
when "0010001011"=> A <= "00010101111001100111111011000001"; -- Index 139 Coefficient 0.17109665335
when "0010001100"=> A <= "00010110001100100110101010001000"; -- Index 140 Coefficient 0.17341357852
when "0010001101"=> A <= "00010110011111101011110101110100"; -- Index 141 Coefficient 0.17574279949
when "0010001110"=> A <= "000101101100101011101011001001"; -- Index 142 Coefficient 0.17808422856
when "0010001111"=> A <= "00010111000110001001010111001001"; -- Index 143 Coefficient 0.18043777757
when "0010010000"=> A <= "000101110110011000011000110110110"; -- Index 144 Coefficient 0.18280335792
when "0010010001"=> A <= "00010111101101000000000111010001"; -- Index 145 Coefficient 0.18518088054
when "0010010010"=> A <= "0001100000000010010011010101001"; -- Index 146 Coefficient 0.18757025593

```

when "0010010011"=> A <= "00011000010100001111101110001110"; -- Index 147 Coefficient 0.18997139412
when "0010010100"=> A <= "00011000101000000000101110101110"; -- Index 148 Coefficient 0.19238420471
when "0010010101"=> A <= "00011000111011110111110011110100"; -- Index 149 Coefficient 0.19480859686
when "0010010110"=> A <= "00011001001111110100111010011110"; -- Index 150 Coefficient 0.19724447930
when "0010010111"=> A <= "00011001100011110111111111100110"; -- Index 151 Coefficient 0.19969176031
when "0010011000"=> A <= "000110011110000000100000000110"; -- Index 152 Coefficient 0.20215034775
when "0010011001"=> A <= "00011010001100001111111000111000"; -- Index 153 Coefficient 0.20462014907
when "0010011010"=> A <= "00011010100000100100100110110100"; -- Index 154 Coefficient 0.20710107127
when "0010011011"=> A <= "00011010110100111111000110110001"; -- Index 155 Coefficient 0.20959302095
when "0010011100"=> A <= "00011011001001011111010101100110"; -- Index 156 Coefficient 0.21209590429
when "0010011101"=> A <= "00011011011110000101010000001001"; -- Index 157 Coefficient 0.21460962706
when "0010011110"=> A <= "00011011110010110000110011001110"; -- Index 158 Coefficient 0.21713409461
when "0010011111"=> A <= "00011100000111100001111011101001"; -- Index 159 Coefficient 0.21966921190
when "0010100000"=> A <= "00011100011100011000100110001101"; -- Index 160 Coefficient 0.22221488349
when "0010100001"=> A <= "00011100110001010100101111101100"; -- Index 161 Coefficient 0.22477101353
when "0010100010"=> A <= "00011101000110010110010100111000"; -- Index 162 Coefficient 0.22733750579
when "0010100011"=> A <= "00011101011011011101010010100010"; -- Index 163 Coefficient 0.22991426364
when "0010100100"=> A <= "00011101110000101001100101011000"; -- Index 164 Coefficient 0.23250119006
when "0010100101"=> A <= "00011110000101111011001010001010"; -- Index 165 Coefficient 0.23509818766
when "0010100110"=> A <= "00011110011011010001111101100101"; -- Index 166 Coefficient 0.23770515866
when "0010100111"=> A <= "00011110110000101101111100011000"; -- Index 167 Coefficient 0.24032200492
when "0010101000"=> A <= "00011111000110001111000011001110"; -- Index 168 Coefficient 0.24294862790
when "0010101001"=> A <= "00011111011011110101001110110011"; -- Index 169 Coefficient 0.24558492873
when "0010101010"=> A <= "00011111110001100000011011110001"; -- Index 170 Coefficient 0.24823080814
when "0010101011"=> A <= "00100000000111010000100110110100"; -- Index 171 Coefficient 0.25088616651
when "0010101100"=> A <= "00100000011101000101101100100100"; -- Index 172 Coefficient 0.25355090389
when "0010101101"=> A <= "00100000110010111111101001101010"; -- Index 173 Coefficient 0.25622491993
when "0010101110"=> A <= "00100001001000111110011010101101"; -- Index 174 Coefficient 0.25890811396
when "0010101111"=> A <= "00100001011111000001111100010101"; -- Index 175 Coefficient 0.26160038497
when "0010110000"=> A <= "00100001110101001010001011001000"; -- Index 176 Coefficient 0.26430163159
when "0010110001"=> A <= "00100010001011010111000011101011"; -- Index 177 Coefficient 0.26701175212
when "0010110010"=> A <= "00100010100001101000100010100100"; -- Index 178 Coefficient 0.26973064452
when "0010110011"=> A <= "00100010110111111110100100010111"; -- Index 179 Coefficient 0.27245820644
when "0010110100"=> A <= "00100011001110011001000101100111"; -- Index 180 Coefficient 0.27519433517
when "0010110101"=> A <= "0010001110010011100000010110110"; -- Index 181 Coefficient 0.27793892771
when "0010110110"=> A <= "00100011111011011011011000101000"; -- Index 182 Coefficient 0.28069188073
when "0010110111"=> A <= "00100100010010000011000011011101"; -- Index 183 Coefficient 0.28345309057
when "0010111000"=> A <= "0010010010100010111011111110110"; -- Index 184 Coefficient 0.28622245328
when "0010111001"=> A <= "00100100111111011111001010010100"; -- Index 185 Coefficient 0.28899986460
when "0010111010"=> A <= "00100101010110010011011111010101"; -- Index 186 Coefficient 0.29178521995
when "0010111011"=> A <= "00100101101101001011111011011000"; -- Index 187 Coefficient 0.29457841447
when "0010111100"=> A <= "00100110000100001000011010111100"; -- Index 188 Coefficient 0.29737934300
when "0010111101"=> A <= "00100110011011001000111010011111"; -- Index 189 Coefficient 0.30018790008
when "0010111110"=> A <= "00100110110010001101010110011100"; -- Index 190 Coefficient 0.30300397997
when "0010111111"=> A <= "0010011100100101010101011010001"; -- Index 191 Coefficient 0.30582747665
when "0011000000"=> A <= "00100111100000100001110101011001"; -- Index 192 Coefficient 0.30865828382
when "0011000001"=> A <= "00100111110111110001110001010000"; -- Index 193 Coefficient 0.31149629489
when "0011000010"=> A <= "00101000001111000101011011001111"; -- Index 194 Coefficient 0.31434140302
when "0011000011"=> A <= "00101000100110011100101111110001"; -- Index 195 Coefficient 0.31719350110
when "0011000100"=> A <= "00101000111101110111101011001111"; -- Index 196 Coefficient 0.32005248173
when "0011000101"=> A <= "00101001010101010110001010000010"; -- Index 197 Coefficient 0.32291823729
when "0011000110"=> A <= "00101001101100111000001000100011"; -- Index 198 Coefficient 0.32579065988
when "0011000111"=> A <= "00101010000100011101100011001000"; -- Index 199 Coefficient 0.32866964134
when "0011001000"=> A <= "00101010011100000110010110001010"; -- Index 200 Coefficient 0.33155507330
when "0011001001"=> A <= "00101010110011110010011101111111"; -- Index 201 Coefficient 0.33444684712
when "0011001010"=> A <= "00101011001011100001110110111110"; -- Index 202 Coefficient 0.33734485392
when "0011001011"=> A <= "00101011100011010100011101011011"; -- Index 203 Coefficient 0.34024898459
when "0011001100"=> A <= "00101011111011001010001101101100"; -- Index 204 Coefficient 0.34315912980

```

when "0011001101"=> A <= "00101100010011000011000100000110"; -- Index 205 Coefficient 0.34607517998
when "0011001110"=> A <= "00101100101010111110111100111101"; -- Index 206 Coefficient 0.34899702534
when "0011001111"=> A <= "00101101000010111101110100100101"; -- Index 207 Coefficient 0.35192455588
when "0011010000"=> A <= "0010110101101011111100111010001"; -- Index 208 Coefficient 0.35485766137
when "0011010001"=> A <= "00101101110011000100010001010100"; -- Index 209 Coefficient 0.35779623139
when "0011010010"=> A <= "00101110001011001011101111000001"; -- Index 210 Coefficient 0.36074015531
when "0011010011"=> A <= "00101110100011010101111100101001"; -- Index 211 Coefficient 0.36368932228
when "0011010100"=> A <= "00101110111011100010110110011101"; -- Index 212 Coefficient 0.36664362126
when "0011010101"=> A <= "00101111010011110010011000110000"; -- Index 213 Coefficient 0.36960294104
when "0011010110"=> A <= "00101111101100000100011111110010"; -- Index 214 Coefficient 0.37256717020
when "0011010111"=> A <= "00110000000100011001000111110011"; -- Index 215 Coefficient 0.37553619713
when "0011011000"=> A <= "0011000001110011000000101000010"; -- Index 216 Coefficient 0.37850991005
when "0011011001"=> A <= "00110000110101001001101011110001"; -- Index 217 Coefficient 0.38148819700
when "0011011010"=> A <= "00110001001101100101100000001101"; -- Index 218 Coefficient 0.38447094586
when "0011011011"=> A <= "00110001100110000011100110100110"; -- Index 219 Coefficient 0.38745804432
when "0011011100"=> A <= "00110001111110100011111011001011"; -- Index 220 Coefficient 0.39044937992
when "0011011101"=> A <= "00110010010111000110011010001000"; -- Index 221 Coefficient 0.39344484004
when "0011011110"=> A <= "00110010101111101010111111101101"; -- Index 222 Coefficient 0.39644431190
when "0011011111"=> A <= "00110011001000010001101000000111"; -- Index 223 Coefficient 0.39944768258
when "0011100000"=> A <= "00110011100000111010001111100010"; -- Index 224 Coefficient 0.40245483899
when "0011100001"=> A <= "00110011111001100100110010001100"; -- Index 225 Coefficient 0.40546566793
when "0011100010"=> A <= "00110100010010010001001100010001"; -- Index 226 Coefficient 0.40848005602
when "0011100011"=> A <= "0011010010101011111011001111110"; -- Index 227 Coefficient 0.41149788979
when "0011100100"=> A <= "00110101000011101111010111011110"; -- Index 228 Coefficient 0.41451905562
when "0011100101"=> A <= "0011010101110010000100000011101"; -- Index 229 Coefficient 0.41754343976
when "0011100110"=> A <= "00110101110101010100010010100111"; -- Index 230 Coefficient 0.42057092833
when "0011100111"=> A <= "00110110001110001001001000101000"; -- Index 231 Coefficient 0.42360140737
when "0011101000"=> A <= "0011011010011011111011111001001"; -- Index 232 Coefficient 0.42663476277
when "0011101001"=> A <= "00110110111111110111010010010110"; -- Index 233 Coefficient 0.42967088033
when "0011101010"=> A <= "00110111011000110000011110011001"; -- Index 234 Coefficient 0.43270964575
when "0011101011"=> A <= "00110111110001101010111111011100"; -- Index 235 Coefficient 0.43575094460
when "0011101100"=> A <= "00111000001010100110110001101010"; -- Index 236 Coefficient 0.43879466240
when "0011101101"=> A <= "00111000100011100011110001001101"; -- Index 237 Coefficient 0.44184068454
when "0011101110"=> A <= "00111000111100100001111010001110"; -- Index 238 Coefficient 0.44488889635
when "0011101111"=> A <= "00111001010101100001001000110111"; -- Index 239 Coefficient 0.44793918306
when "0011110000"=> A <= "00111001101110100001011001010001"; -- Index 240 Coefficient 0.45099142984
when "0011110001"=> A <= "0011101000011100010100111100101"; -- Index 241 Coefficient 0.45404552175
when "0011110010"=> A <= "0011101010000010010010111111101"; -- Index 242 Coefficient 0.45710134383
when "0011110011"=> A <= "00111010111001100111101110100010"; -- Index 243 Coefficient 0.46015878101
when "0011110100"=> A <= "00111011010010101010111110110111"; -- Index 244 Coefficient 0.46321771820
when "0011110101"=> A <= "0011101110101110111111110110011"; -- Index 245 Coefficient 0.46627804022
when "0011110110"=> A <= "00111100000100110101001000110001"; -- Index 246 Coefficient 0.46933963185
when "0011110111"=> A <= "0011110001110111010111001011110"; -- Index 247 Coefficient 0.47240237783
when "0011111000"=> A <= "00111100110111000001001101000010"; -- Index 248 Coefficient 0.47546616284
when "0011111001"=> A <= "0011110101000000011111111100110"; -- Index 249 Coefficient 0.47853087153
when "0011111010"=> A <= "00111101101001001111001101010001"; -- Index 250 Coefficient 0.48159638853
when "0011111011"=> A <= "00111110000010010110110010001101"; -- Index 251 Coefficient 0.48466259841
when "0011111100"=> A <= "00111110011011011110101010100001"; -- Index 252 Coefficient 0.48772938574
when "0011111101"=> A <= "00111110110100100110110010010100"; -- Index 253 Coefficient 0.49079663505
when "0011111110"=> A <= "00111111001101101111000101110000"; -- Index 254 Coefficient 0.49386423086
when "0011111111"=> A <= "0011111110011011011100000111100"; -- Index 255 Coefficient 0.49693205768
when "0100000000"=> A <= "01000000000000000000000000000000"; -- Index 256 Coefficient 0.50000000000
when "0100000001"=> A <= "01000000011001001000011111000100"; -- Index 257 Coefficient 0.50306794232
when "0100000010"=> A <= "01000000110010010000111010010000"; -- Index 258 Coefficient 0.50613576914
when "0100000011"=> A <= "01000001001011011001001101101100"; -- Index 259 Coefficient 0.50920336495
when "0100000100"=> A <= "01000001100100100001010101011111"; -- Index 260 Coefficient 0.51227061426
when "0100000101"=> A <= "01000001111101101001001101110011"; -- Index 261 Coefficient 0.51533740159
when "0100000110"=> A <= "01000010010110110000110010101111"; -- Index 262 Coefficient 0.51840361147

```

```

when "0100000111"=> A <= "0100001010111111000000000011010"; -- Index 263 Coefficient 0.52146912847
when "0100001000"=> A <= "01000011001000111110110010111110"; -- Index 264 Coefficient 0.52453383716
when "0100001001"=> A <= "01000011100010000101000110100010"; -- Index 265 Coefficient 0.52759762217
when "0100001010"=> A <= "01000011111011001010110111001111"; -- Index 266 Coefficient 0.53066036815
when "0100001011"=> A <= "01000100010100010000000001001101"; -- Index 267 Coefficient 0.53372195978
when "0100001100"=> A <= "01000100101101010100100000100101"; -- Index 268 Coefficient 0.53678228180
when "0100001101"=> A <= "01000101000110011000010001011110"; -- Index 269 Coefficient 0.53984121899
when "0100001110"=> A <= "01000101011111011011010000000011"; -- Index 270 Coefficient 0.54289865617
when "0100001111"=> A <= "01000101111000011101011000011011"; -- Index 271 Coefficient 0.54595447825
when "0100010000"=> A <= "01000110010001011110100110101111"; -- Index 272 Coefficient 0.54900857016
when "0100010001"=> A <= "01000110101010011110110111001001"; -- Index 273 Coefficient 0.55206081694
when "0100010010"=> A <= "01000111000011011110000101110010"; -- Index 274 Coefficient 0.55511110365
when "0100010011"=> A <= "01000111011100011100001110110011"; -- Index 275 Coefficient 0.55815931546
when "0100010100"=> A <= "0100011110101011001001110010110"; -- Index 276 Coefficient 0.56120533760
when "0100010101"=> A <= "01001000001110010101000000100100"; -- Index 277 Coefficient 0.56424905540
when "0100010110"=> A <= "01001000100111001111100001100111"; -- Index 278 Coefficient 0.56729035425
when "0100010111"=> A <= "0100100100000001000101101101010"; -- Index 279 Coefficient 0.57032911967
when "0100100000"=> A <= "01001001011001000000100000110111"; -- Index 280 Coefficient 0.57336523723
when "0100100001"=> A <= "01001001110001110110110111011000"; -- Index 281 Coefficient 0.57639859263
when "0100101000"=> A <= "010010100010101010111011011001"; -- Index 282 Coefficient 0.57942907167
when "0100101001"=> A <= "0100101010001101111011111000011"; -- Index 283 Coefficient 0.58245656024
when "0100101100"=> A <= "01001010111100010000101000100010"; -- Index 284 Coefficient 0.58548094438
when "0100101101"=> A <= "01001011010101000000100110000010"; -- Index 285 Coefficient 0.58850211021
when "0100101110"=> A <= "01001011101101101110110011101111"; -- Index 286 Coefficient 0.59151994398
when "0100101111"=> A <= "010011000000110011011001101110100"; -- Index 287 Coefficient 0.59453433207
when "0100100000"=> A <= "01001100011111000101110000011110"; -- Index 288 Coefficient 0.59754516101
when "0100100001"=> A <= "01001100110111101110010111111001"; -- Index 289 Coefficient 0.60055231742
when "0100100010"=> A <= "0100110101000001010100000010011"; -- Index 290 Coefficient 0.60355568810
when "0100100011"=> A <= "01001101101000111001100101111000"; -- Index 291 Coefficient 0.60655515996
when "0100100100"=> A <= "01001110000001011100000100110101"; -- Index 292 Coefficient 0.60955062008
when "0100100101"=> A <= "01001110011001111100011001011010"; -- Index 293 Coefficient 0.61254195568
when "0100100110"=> A <= "01001110110010011010011111110011"; -- Index 294 Coefficient 0.61552905414
when "0100100111"=> A <= "01001111001010110110010100001111"; -- Index 295 Coefficient 0.61851180300
when "0100101000"=> A <= "01001111100011001111110010111110"; -- Index 296 Coefficient 0.62149008995
when "0100101001"=> A <= "0100111111011100110111000001101"; -- Index 297 Coefficient 0.62446380287
when "0100101010"=> A <= "01010000010011111011100000001110"; -- Index 298 Coefficient 0.62743282980
when "0100101011"=> A <= "010100000101100001101100111010000"; -- Index 299 Coefficient 0.63039705896
when "0100101100"=> A <= "01010001000100011101001001100011"; -- Index 300 Coefficient 0.63335637874
when "0100101101"=> A <= "01010001011100101010000011010111"; -- Index 301 Coefficient 0.63631067772
when "0100101110"=> A <= "01010001110100110100010000111111"; -- Index 302 Coefficient 0.63925984469
when "0100101111"=> A <= "01010010001100111011101110101100"; -- Index 303 Coefficient 0.64220376861
when "0100110000"=> A <= "01010010100101000000011000101111"; -- Index 304 Coefficient 0.64514233863
when "0100110001"=> A <= "01010010111101000010001011011011"; -- Index 305 Coefficient 0.64807544412
when "0100110010"=> A <= "01010011010101000001000011000011"; -- Index 306 Coefficient 0.65100297466
when "0100110011"=> A <= "0101001110110011110011101111010"; -- Index 307 Coefficient 0.65392482002
when "0100110100"=> A <= "01010100000100110101110010010100"; -- Index 308 Coefficient 0.65684087020
when "0100110101"=> A <= "01010100011100101011100010100101"; -- Index 309 Coefficient 0.65975101541
when "0100110110"=> A <= "01010100110100011110001001000010"; -- Index 310 Coefficient 0.66265514608
when "0100110111"=> A <= "01010101001100001101100010000001"; -- Index 311 Coefficient 0.66555315288
when "0100111000"=> A <= "01010101100011111001101001110110"; -- Index 312 Coefficient 0.66844492670
when "0100111001"=> A <= "01010101111011100010011100111000"; -- Index 313 Coefficient 0.67133035866
when "0100111010"=> A <= "01010110010011000111110111011101"; -- Index 314 Coefficient 0.67420934012
when "0100111011"=> A <= "01010110101010101001110101111110"; -- Index 315 Coefficient 0.67708176271
when "0100111100"=> A <= "010101110000100010000101000110001"; -- Index 316 Coefficient 0.67994751827
when "0100111101"=> A <= "01010111011001100011010000001111"; -- Index 317 Coefficient 0.68280649890
when "0100111110"=> A <= "01010111110000111010100100110001"; -- Index 318 Coefficient 0.68565859698
when "0100111111"=> A <= "01011000001000001110001110110000"; -- Index 319 Coefficient 0.68850370511
when "0101000000"=> A <= "01011000011111011110001010100111"; -- Index 320 Coefficient 0.69134171618

```

when "0101000001"=> A <= "01011000110110101010010100101111"; -- Index 321 Coefficient 0.69417252335
when "0101000010"=> A <= "01011001001101110010101001100100"; -- Index 322 Coefficient 0.69699602003
when "0101000011"=> A <= "01011001100100110111000101100001"; -- Index 323 Coefficient 0.69981209992
when "0101000100"=> A <= "01011001111011110111100101000100"; -- Index 324 Coefficient 0.70262065700
when "0101000101"=> A <= "01011010010010110100000100101000"; -- Index 325 Coefficient 0.70542158553
when "0101000110"=> A <= "01011010101001101100100000101011"; -- Index 326 Coefficient 0.70821478005
when "0101000111"=> A <= "01011011000000100000110101101100"; -- Index 327 Coefficient 0.71100013540
when "0101001000"=> A <= "0101101101011101000100000001010"; -- Index 328 Coefficient 0.71377754672
when "0101001001"=> A <= "01011011101101111100111100100011"; -- Index 329 Coefficient 0.71654690943
when "0101001010"=> A <= "01011100000100100100100111011000"; -- Index 330 Coefficient 0.71930811927
when "0101001011"=> A <= "01011100011011000111111101001010"; -- Index 331 Coefficient 0.72206107229
when "0101001100"=> A <= "01011100110001100110111010011001"; -- Index 332 Coefficient 0.72480566483
when "0101001101"=> A <= "0101110100100000001011011101001"; -- Index 333 Coefficient 0.72754179356
when "0101001110"=> A <= "01011101011110010111011101011100"; -- Index 334 Coefficient 0.73026935548
when "0101001111"=> A <= "01011101110100101000111100010101"; -- Index 335 Coefficient 0.73298824788
when "0101010000"=> A <= "01011110001010110101110100111000"; -- Index 336 Coefficient 0.73569836841
when "0101010001"=> A <= "01011110100000111110000011101011"; -- Index 337 Coefficient 0.73839961503
when "0101010010"=> A <= "01011110110111000001100101010011"; -- Index 338 Coefficient 0.74109188604
when "0101010011"=> A <= "01011111001101000000010110010110"; -- Index 339 Coefficient 0.74377508007
when "0101010100"=> A <= "01011111100010111010010011011100"; -- Index 340 Coefficient 0.74644909611
when "0101010101"=> A <= "0101111111000101111011001001100"; -- Index 341 Coefficient 0.74911383349
when "0101010110"=> A <= "0110000000111001111100100001111"; -- Index 342 Coefficient 0.75176919186
when "0101010111"=> A <= "01100000100100001010110001001101"; -- Index 343 Coefficient 0.75441507127
when "0101011000"=> A <= "01100000111001110000111100110010"; -- Index 344 Coefficient 0.75705137210
when "0101011001"=> A <= "01100001001111010010000011101000"; -- Index 345 Coefficient 0.75967799508
when "0101011010"=> A <= "01100001100100101110000010011011"; -- Index 346 Coefficient 0.76229484134
when "0101011011"=> A <= "01100001111010000100110101110110"; -- Index 347 Coefficient 0.76490181234
when "0101011100"=> A <= "01100010001111010110011010101000"; -- Index 348 Coefficient 0.76749880994
when "0101011101"=> A <= "011000101001001000010101101011110"; -- Index 349 Coefficient 0.77008573636
when "0101011110"=> A <= "01100010111001101001101011001000"; -- Index 350 Coefficient 0.77266249421
when "0101011111"=> A <= "01100011001110101011010000010100"; -- Index 351 Coefficient 0.77522898647
when "0101100000"=> A <= "01100011100011100111011001110011"; -- Index 352 Coefficient 0.77778511651
when "0101100001"=> A <= "01100011111000011110000100010111"; -- Index 353 Coefficient 0.78033078810
when "0101100010"=> A <= "01100100001101001111001100110010"; -- Index 354 Coefficient 0.78286590539
when "0101100011"=> A <= "01100100100001111010101111110111"; -- Index 355 Coefficient 0.78539037294
when "0101100100"=> A <= "01100100110110100000101010011010"; -- Index 356 Coefficient 0.78790409571
when "0101100101"=> A <= "01100101001011000000111001001111"; -- Index 357 Coefficient 0.79040697905
when "0101100110"=> A <= "01100101011111011011011001001100"; -- Index 358 Coefficient 0.79289892873
when "0101100111"=> A <= "01100101110011110000000111001000"; -- Index 359 Coefficient 0.79537985093
when "0101101000"=> A <= "0110011000011111110111111111010"; -- Index 360 Coefficient 0.79784965225
when "0101101001"=> A <= "0110011001110000100000000011010"; -- Index 361 Coefficient 0.80030823969
when "0101101010"=> A <= "01100110110000001011000101100010"; -- Index 362 Coefficient 0.80275552070
when "0101101011"=> A <= "01100111000100001000001100001100"; -- Index 363 Coefficient 0.80519140314
when "0101101100"=> A <= "01100111010111111111010001010010"; -- Index 364 Coefficient 0.80761579529
when "0101101101"=> A <= "01100111101011110000010001110010"; -- Index 365 Coefficient 0.81002860588
when "0101101110"=> A <= "01100111111111011011001010100111"; -- Index 366 Coefficient 0.81242974407
when "0101101111"=> A <= "01101000010010111111111000101111"; -- Index 367 Coefficient 0.81481911946
when "0101110000"=> A <= "01101000100110011110011001001010"; -- Index 368 Coefficient 0.81719664208
when "0101110001"=> A <= "01101000111001110110101000110111"; -- Index 369 Coefficient 0.81956222243
when "0101110010"=> A <= "01101001001101001000100100110111"; -- Index 370 Coefficient 0.82191577144
when "0101110011"=> A <= "01101001100000010100001010001100"; -- Index 371 Coefficient 0.82425720051
when "0101110100"=> A <= "01101001110011011001010101111000"; -- Index 372 Coefficient 0.82658642148
when "0101110101"=> A <= "01101010000110011000000100111111"; -- Index 373 Coefficient 0.82890334665
when "0101110110"=> A <= "01101010011001010000010100100101"; -- Index 374 Coefficient 0.83120788880
when "0101110111"=> A <= "01101010101100000010000001110001"; -- Index 375 Coefficient 0.83349996115
when "0101111000"=> A <= "0110101011111010101001001101001"; -- Index 376 Coefficient 0.83577947742
when "0101111001"=> A <= "0110101101000101000101001010101"; -- Index 377 Coefficient 0.83804635179
when "0101111010"=> A <= "011010111000111011110110111101"; -- Index 378 Coefficient 0.84030049890

```

when "0101111011"=> A <= "01101011110110000110100100101011"; -- Index 379 Coefficient 0.84254183389
when "0101111100"=> A <= "01101100001000010110111010101010"; -- Index 380 Coefficient 0.84477027237
when "0101111101"=> A <= "01101100011010100000011101000110"; -- Index 381 Coefficient 0.84698573044
when "0101111110"=> A <= "01101100101100100011001001001100"; -- Index 382 Coefficient 0.84918812470
when "0101111111"=> A <= "01101100111110011110111100001001"; -- Index 383 Coefficient 0.85137737223
when "0110000000"=> A <= "01101101010000010011110011001101"; -- Index 384 Coefficient 0.85355339059
when "0110000001"=> A <= "01101101100010000001101011101000"; -- Index 385 Coefficient 0.85571609787
when "0110000010"=> A <= "01101101110011101000100010101010"; -- Index 386 Coefficient 0.85786541264
when "0110000011"=> A <= "01101110000101001000010101100110"; -- Index 387 Coefficient 0.86000125398
when "0110000100"=> A <= "01101110010110100001000001110000"; -- Index 388 Coefficient 0.86212354148
when "0110000101"=> A <= "01101110100111110010100100011011"; -- Index 389 Coefficient 0.86423219522
when "0110000110"=> A <= "01101110111000111100111010111110"; -- Index 390 Coefficient 0.86632713584
when "0110000111"=> A <= "01101111001010000000000010101111"; -- Index 391 Coefficient 0.86840828444
when "0110001000"=> A <= "01101111011010111011111001000101"; -- Index 392 Coefficient 0.87047556268
when "0110001001"=> A <= "01101111101011110000011011011010"; -- Index 393 Coefficient 0.87252889272
when "0110001010"=> A <= "0110111111100011101100111000111"; -- Index 394 Coefficient 0.87456819726
when "0110001011"=> A <= "01110000001101000011011001100111"; -- Index 395 Coefficient 0.87659339952
when "0110001100"=> A <= "01110000011101100001110000011000"; -- Index 396 Coefficient 0.87860442325
when "0110001101"=> A <= "01110000101101111000101000110110"; -- Index 397 Coefficient 0.88060119274
when "0110001110"=> A <= "01110000111110001000000000111111"; -- Index 398 Coefficient 0.88258363281
when "0110001111"=> A <= "01110001001110001111110100110101"; -- Index 399 Coefficient 0.88455166882
when "0110010000"=> A <= "01110001011110010000000011010110"; -- Index 400 Coefficient 0.88650522668
when "0110010001"=> A <= "01110001101110001000101001100110"; -- Index 401 Coefficient 0.88844423284
when "0110010010"=> A <= "01110001111101111001100101001000"; -- Index 402 Coefficient 0.89036861429
when "0110010011"=> A <= "01110010001101100010110011100000"; -- Index 403 Coefficient 0.89227829858
when "0110010100"=> A <= "01110010011101000100010010010011"; -- Index 404 Coefficient 0.89417321381
when "0110010101"=> A <= "0111001010110001110111111001001"; -- Index 405 Coefficient 0.89605328865
when "0110010110"=> A <= "01110010111011101111110111101010"; -- Index 406 Coefficient 0.89791845230
when "0110010111"=> A <= "01110011001010111001111001011110"; -- Index 407 Coefficient 0.89976863455
when "0110011000"=> A <= "0111001101100111110000010010000"; -- Index 408 Coefficient 0.90160376574
when "0110011001"=> A <= "0111001110100011011000111101100"; -- Index 409 Coefficient 0.90342377677
when "0110011010"=> A <= "01110011110111101000011111011110"; -- Index 410 Coefficient 0.90522859913
when "0110011011"=> A <= "01110100000110010010101111010101"; -- Index 411 Coefficient 0.90701816485
when "0110011100"=> A <= "01110100010100110100111101000001"; -- Index 412 Coefficient 0.90879240658
when "0110011101"=> A <= "01110100100011001111000110010000"; -- Index 413 Coefficient 0.91055125750
when "0110011110"=> A <= "01110100110001100001001000110110"; -- Index 414 Coefficient 0.91229465139
when "0110011111"=> A <= "01110100111111101011000010100101"; -- Index 415 Coefficient 0.91402252263
when "0110100000"=> A <= "01110101001101101100110001010010"; -- Index 416 Coefficient 0.91573480615
when "0110100001"=> A <= "01110101011011100110010010110010"; -- Index 417 Coefficient 0.91743143749
when "0110100010"=> A <= "01110101101001010111100100111100"; -- Index 418 Coefficient 0.91911235278
when "0110100011"=> A <= "01110101110111000000100101101000"; -- Index 419 Coefficient 0.92077748872
when "0110100100"=> A <= "01110110000100100001010010110000"; -- Index 420 Coefficient 0.92242678262
when "0110100101"=> A <= "011101100100011110011010100001110"; -- Index 421 Coefficient 0.92406017240
when "0110100110"=> A <= "01110110011111001001101001111110"; -- Index 422 Coefficient 0.92567759655
when "0110100111"=> A <= "0111011010110001000100111111101"; -- Index 423 Coefficient 0.92727899418
when "0110101000"=> A <= "01110110111001010000011010001010"; -- Index 424 Coefficient 0.92886430500
when "0110101001"=> A <= "01110111000110000111000110100101"; -- Index 425 Coefficient 0.93043346932
when "0110101010"=> A <= "01110111010010110101010011001110"; -- Index 426 Coefficient 0.93198642806
when "0110101011"=> A <= "0111011101111011010111110001001"; -- Index 427 Coefficient 0.93352312276
when "0110101100"=> A <= "01110111101011111000000101011001"; -- Index 428 Coefficient 0.93504349555
when "0110101101"=> A <= "01110111111000001100100111000011"; -- Index 429 Coefficient 0.93654748921
when "0110101110"=> A <= "01111000000100011000100001001101"; -- Index 430 Coefficient 0.93803504710
when "0110101111"=> A <= "01111000010000011011110001111111"; -- Index 431 Coefficient 0.93950611321
when "0110110000"=> A <= "01111000011100010110010111100011"; -- Index 432 Coefficient 0.94096063217
when "0110110001"=> A <= "01111000101000001000010000000010"; -- Index 433 Coefficient 0.94239854922
when "0110110010"=> A <= "01111000110011110001011000101001"; -- Index 434 Coefficient 0.94381981020
when "0110110011"=> A <= "01111000111111010001110010100100"; -- Index 435 Coefficient 0.94522436162
when "0110110100"=> A <= "01111001001010101001011001000010"; -- Index 436 Coefficient 0.94661215060

```


when "0110110101"=> A <= "01111001010101111000001011010011"; -- Index 437 Coefficient 0.94798312488
when "0110110110"=> A <= "01111001100000111110000111101000"; -- Index 438 Coefficient 0.94933723285
when "0110110111"=> A <= "0111100110101111101001100010011"; -- Index 439 Coefficient 0.95067442352
when "0110111000"=> A <= "01111001110110101111010111101000"; -- Index 440 Coefficient 0.95199464656
when "0110111001"=> A <= "01111010000001011010100111111011"; -- Index 441 Coefficient 0.95329785226
when "0110111010"=> A <= "011110100010111110011101110101000"; -- Index 442 Coefficient 0.95458399155
when "0110111011"=> A <= "01111010010110010110010001000010"; -- Index 443 Coefficient 0.95585301600
when "0110111100"=> A <= "01111010100000100110100110100011"; -- Index 444 Coefficient 0.95710487785
when "0110111101"=> A <= "01111010101010101010111010100110"; -- Index 445 Coefficient 0.95833952996
when "0110111110"=> A <= "01111010110100101100001011101000"; -- Index 446 Coefficient 0.95955692585
when "0110111111"=> A <= "0111101011110100001011000000101"; -- Index 447 Coefficient 0.96075701967
when "0111000000"=> A <= "0111101100100000101011110011110"; -- Index 448 Coefficient 0.96193976626
when "0111000001"=> A <= "01111011010001110000011101010011"; -- Index 449 Coefficient 0.96310512107
when "0111000010"=> A <= "01111011011011001010010011000100"; -- Index 450 Coefficient 0.96425304024
when "0111000011"=> A <= "01111011100100011010111110010111"; -- Index 451 Coefficient 0.96538348054
when "0111000100"=> A <= "01111011101010100010011101101110"; -- Index 452 Coefficient 0.96649639942
when "0111000101"=> A <= "0111101111011010000010111110000"; -- Index 453 Coefficient 0.96759175497
when "0111000110"=> A <= "0111101111111010101110011000100"; -- Index 454 Coefficient 0.96866950596
when "0111000111"=> A <= "01111100001000000001100110010100"; -- Index 455 Coefficient 0.96972961180
when "0111001000"=> A <= "01111100010000100100001000001010"; -- Index 456 Coefficient 0.97077203259
when "0111001001"=> A <= "0111110001100011101010111010001"; -- Index 457 Coefficient 0.97179672908
when "0111001010"=> A <= "0111110010000100101010010010110"; -- Index 458 Coefficient 0.97280366269
when "0111001011"=> A <= "01111100101001010011111000001001"; -- Index 459 Coefficient 0.97379279551
when "0111001100"=> A <= "01111100110001010001000111011001"; -- Index 460 Coefficient 0.97476409030
when "0111001101"=> A <= "01111100111001000100111110110111"; -- Index 461 Coefficient 0.97571751048
when "0111001110"=> A <= "0111110100000010111101101010111"; -- Index 462 Coefficient 0.97665302018
when "0111001111"=> A <= "01111101001000010000100001101100"; -- Index 463 Coefficient 0.97757058415
when "0111010000"=> A <= "01111101001111101000001010101110"; -- Index 464 Coefficient 0.97847016787
when "0111010001"=> A <= "01111101010101010110010111010010"; -- Index 465 Coefficient 0.97935173745
when "0111010010"=> A <= "0111110101110111011000110010010"; -- Index 466 Coefficient 0.98021525971
when "0111010011"=> A <= "01111101100100101101001010101000"; -- Index 467 Coefficient 0.98106070213
when "0111010100"=> A <= "01111101101011101000000111001111"; -- Index 468 Coefficient 0.98188803290
when "0111010101"=> A <= "01111101110010010000010111000101"; -- Index 469 Coefficient 0.98269722085
when "0111010110"=> A <= "01111101111000101111000101001000"; -- Index 470 Coefficient 0.98348823552
when "0111010111"=> A <= "0111110111111000100010000011000"; -- Index 471 Coefficient 0.98426104714
when "0111011000"=> A <= "0111111000010100111110111110111"; -- Index 472 Coefficient 0.98501562660
when "0111011001"=> A <= "01111110001011010001111010101000"; -- Index 473 Coefficient 0.98575194549
when "0111011010"=> A <= "01111110010001001010010111101111"; -- Index 474 Coefficient 0.98646997610
when "0111011011"=> A <= "01111110010101011001001110010010"; -- Index 475 Coefficient 0.98716969139
when "0111011100"=> A <= "01111110011100011110011101011001"; -- Index 476 Coefficient 0.98785106502
when "0111011101"=> A <= "01111110100001111010000100001100"; -- Index 477 Coefficient 0.98851407133
when "0111011110"=> A <= "01111110100111001100000001110110"; -- Index 478 Coefficient 0.98915868536
when "0111011111"=> A <= "01111110101100010100010101100011"; -- Index 479 Coefficient 0.98978488284
when "0111100000"=> A <= "01111110110001010010111110100000"; -- Index 480 Coefficient 0.99039264020
when "0111100001"=> A <= "0111111011011000001111101111100"; -- Index 481 Coefficient 0.99098193455
when "0111100010"=> A <= "01111110111010110011001101000111"; -- Index 482 Coefficient 0.99155274372
when "0111100011"=> A <= "0111111011111010100110001010100"; -- Index 483 Coefficient 0.99210504619
when "0111100100"=> A <= "01111111000011101100100111110101"; -- Index 484 Coefficient 0.99263882119
when "0111100101"=> A <= "0111111100011111010101111111111"; -- Index 485 Coefficient 0.99315404862
when "0111100110"=> A <= "0111111100101111111001001001010"; -- Index 486 Coefficient 0.99365070908
when "0111100111"=> A <= "0111111100111111001110010101011"; -- Index 487 Coefficient 0.99412878387
when "0111101000"=> A <= "01111111010011101010101011111110"; -- Index 488 Coefficient 0.99458825498
when "0111101001"=> A <= "01111111010111010001110100011101"; -- Index 489 Coefficient 0.99502910513
when "0111101010"=> A <= "0111111101101010101111001011100011"; -- Index 490 Coefficient 0.99545131771
when "0111101011"=> A <= "0111111101110000010110000110000"; -- Index 491 Coefficient 0.99585487683
when "0111101100"=> A <= "01111111100001001100100011100010"; -- Index 492 Coefficient 0.99623976730
when "0111101101"=> A <= "01111111100100001100100011011010"; -- Index 493 Coefficient 0.99660597462
when "0111101110"=> A <= "0111111110011100001010111111011"; -- Index 494 Coefficient 0.99695348500

when "0111101111"=> A <= "01111111101001101111001000101000"; -- Index 495 Coefficient 0.99728228537
when "0111110000"=> A <= "01111111101100010001101101001000"; -- Index 496 Coefficient 0.99759236334
when "0111110001"=> A <= "01111111101110101010011101000000"; -- Index 497 Coefficient 0.99788370723
when "0111110010"=> A <= "01111111110000111001010111111001"; -- Index 498 Coefficient 0.99815630609
when "0111110011"=> A <= "01111111110010111110011101011110"; -- Index 499 Coefficient 0.99841014965
when "0111110100"=> A <= "011111111101001110011011010101010"; -- Index 500 Coefficient 0.99864522834
when "0111110101"=> A <= "01111111110110101011000111011001"; -- Index 501 Coefficient 0.99886153332
when "0111110110"=> A <= "01111111111000010010101011001011"; -- Index 502 Coefficient 0.99905905645
when "0111110111"=> A <= "01111111111001110000011000011111"; -- Index 503 Coefficient 0.99923779029
when "0111111000"=> A <= "01111111111011000100001111000111"; -- Index 504 Coefficient 0.99939772810
when "0111111001"=> A <= "0111111111100001110001110110110"; -- Index 505 Coefficient 0.99953886388
when "0111111010"=> A <= "0111111111101001110010111100000"; -- Index 506 Coefficient 0.99966119229
when "0111111011"=> A <= "0111111111110000100101000111100"; -- Index 507 Coefficient 0.99976470875
when "0111111100"=> A <= "0111111111110110001000011000001"; -- Index 508 Coefficient 0.99984940935
when "0111111101"=> A <= "0111111111111010011100101101001"; -- Index 509 Coefficient 0.99991529090
when "0111111110"=> A <= "0111111111111101100010000101101"; -- Index 510 Coefficient 0.99996235092
when "0111111111"=> A <= "01111111111111011000100001011"; -- Index 511 Coefficient 0.99999058764
when "1000000000"=> A <= "01111111111111111111111111111111"; -- Index 512 Coefficient 1.00000000000
when "1000000001"=> A <= "0111111111111111111011000100001011"; -- Index 513 Coefficient 0.99999058764
when "1000000010"=> A <= "0111111111111111101100010000101101"; -- Index 514 Coefficient 0.99996235092
when "1000000011"=> A <= "011111111111111010011100101101001"; -- Index 515 Coefficient 0.99991529090
when "1000000100"=> A <= "01111111111110110001000011000001"; -- Index 516 Coefficient 0.99984940935
when "1000000101"=> A <= "0111111111110000100101000111100"; -- Index 517 Coefficient 0.99976470875
when "1000000110"=> A <= "0111111111101001110010111100000"; -- Index 518 Coefficient 0.99966119229
when "1000000111"=> A <= "0111111111100001110001110110110"; -- Index 519 Coefficient 0.99953886388
when "1000001000"=> A <= "0111111111101100010000111100011"; -- Index 520 Coefficient 0.99939772810
when "1000001001"=> A <= "01111111111001110000011000011111"; -- Index 521 Coefficient 0.99923779029
when "1000001010"=> A <= "01111111111000010010101011001011"; -- Index 522 Coefficient 0.99905905645
when "1000001011"=> A <= "01111111110110101011000111011001"; -- Index 523 Coefficient 0.99886153332
when "1000001100"=> A <= "01111111110100111001101101011010"; -- Index 524 Coefficient 0.99864522834
when "1000001101"=> A <= "01111111110010111111001101011110"; -- Index 525 Coefficient 0.99841014965
when "1000001110"=> A <= "01111111110000111001010111111001"; -- Index 526 Coefficient 0.99815630609
when "1000001111"=> A <= "011111111011101010100111101000000"; -- Index 527 Coefficient 0.99788370723
when "1000010000"=> A <= "01111111101100010001101101001000"; -- Index 528 Coefficient 0.99759236334
when "1000010001"=> A <= "01111111101001101111001000101000"; -- Index 529 Coefficient 0.99728228537
when "1000010010"=> A <= "01111111100111000010101111111011"; -- Index 530 Coefficient 0.99695348500
when "1000010011"=> A <= "01111111100100001100100011011010"; -- Index 531 Coefficient 0.99660597462
when "1000010100"=> A <= "01111111100001001100100011100010"; -- Index 532 Coefficient 0.99623976730
when "1000010101"=> A <= "01111111101110000010110000110000"; -- Index 533 Coefficient 0.99585487683
when "1000010110"=> A <= "01111111101101010101111001011100011"; -- Index 534 Coefficient 0.99545131771
when "1000010111"=> A <= "0111111110101111010001110100011101"; -- Index 535 Coefficient 0.99502910513
when "1000011000"=> A <= "011111111010011101010101011111110"; -- Index 536 Coefficient 0.99458825498
when "1000011001"=> A <= "011111111001111111001110010101011"; -- Index 537 Coefficient 0.99412878387
when "1000011010"=> A <= "011111111001011111111001001001010"; -- Index 538 Coefficient 0.99365070908
when "1000011011"=> A <= "01111111100011111101010111111111"; -- Index 539 Coefficient 0.99315404862
when "1000011100"=> A <= "011111111000011101100100111110101"; -- Index 540 Coefficient 0.99263882119
when "1000011101"=> A <= "01111110111111010100110001010100"; -- Index 541 Coefficient 0.99210504619
when "1000011110"=> A <= "011111101111010110011001101000111"; -- Index 542 Coefficient 0.99155274372
when "1000011111"=> A <= "01111110110110000111111011111100"; -- Index 543 Coefficient 0.99098193455
when "1000100000"=> A <= "01111110110001010010111110100000"; -- Index 544 Coefficient 0.99039264020
when "1000100001"=> A <= "01111110101100001010001010100011"; -- Index 545 Coefficient 0.98978488284
when "1000100010"=> A <= "01111110100111001100000001110110"; -- Index 546 Coefficient 0.98915868536
when "1000100011"=> A <= "01111110100001111010000100001100"; -- Index 547 Coefficient 0.98851407133
when "1000100100"=> A <= "01111110011100011110011101011001"; -- Index 548 Coefficient 0.98785106502
when "1000100101"=> A <= "0111111001010111001001110010010"; -- Index 549 Coefficient 0.98716969139
when "1000100110"=> A <= "01111110010001001010010111101111"; -- Index 550 Coefficient 0.98646997610
when "1000100111"=> A <= "01111110001011010000111010101000"; -- Index 551 Coefficient 0.98575194549
when "1000101000"=> A <= "01111110000101001111110111110111"; -- Index 552 Coefficient 0.98501562660

```

when "1000101001"=> A <= "0111110111111000100010000011000"; -- Index 553 Coefficient 0.98426104714
when "1000101010"=> A <= "01111101111000101111000101001000"; -- Index 554 Coefficient 0.98348823552
when "1000101011"=> A <= "01111101110010010000010111000101"; -- Index 555 Coefficient 0.98269722085
when "1000101100"=> A <= "01111101101011101000000111001111"; -- Index 556 Coefficient 0.98188803290
when "1000101101"=> A <= "01111101100100110110010110101000"; -- Index 557 Coefficient 0.98106070213
when "1000101110"=> A <= "01111101011101111011000110010010"; -- Index 558 Coefficient 0.98021525971
when "1000101111"=> A <= "01111101010110110110010111010010"; -- Index 559 Coefficient 0.97935173745
when "1000110000"=> A <= "01111101001111101000001010101110"; -- Index 560 Coefficient 0.97847016787
when "1000110001"=> A <= "0111110100100001000010000101100"; -- Index 561 Coefficient 0.97757058415
when "1000110010"=> A <= "0111110100000010111101101010111"; -- Index 562 Coefficient 0.97665302018
when "1000110011"=> A <= "0111110011100100010011111010111"; -- Index 563 Coefficient 0.97571751048
when "1000110100"=> A <= "01111100110001010001000111011001"; -- Index 564 Coefficient 0.97476409030
when "1000110101"=> A <= "01111100101001010011111000001001"; -- Index 565 Coefficient 0.97379279551
when "1000110110"=> A <= "01111100100001001101010010010110"; -- Index 566 Coefficient 0.97280366269
when "1000110111"=> A <= "01111100011000111101010111010001"; -- Index 567 Coefficient 0.97179672908
when "1000111000"=> A <= "01111100010000100100001000001010"; -- Index 568 Coefficient 0.97077203259
when "1000111001"=> A <= "0111110000100000001100110010100"; -- Index 569 Coefficient 0.96972961180
when "1000111010"=> A <= "011111011111111010101110011000100"; -- Index 570 Coefficient 0.96866950596
when "1000111011"=> A <= "011111011110110100000101111110000"; -- Index 571 Coefficient 0.96759175497
when "1000111100"=> A <= "011111011101101100010011101101110"; -- Index 572 Coefficient 0.96649639942
when "1000111101"=> A <= "011111011100100011010111110010111"; -- Index 573 Coefficient 0.96538348054
when "1000111110"=> A <= "011111011011011001010010011000100"; -- Index 574 Coefficient 0.96425304024
when "1000111111"=> A <= "011111011010001110000011101010011"; -- Index 575 Coefficient 0.96310512107
when "1001000000"=> A <= "011111011001000001101011110011110"; -- Index 576 Coefficient 0.96193976626
when "1001000001"=> A <= "011111010111110100001011000000101"; -- Index 577 Coefficient 0.96075701967
when "1001000010"=> A <= "011111010110100101100001011101000"; -- Index 578 Coefficient 0.95955692585
when "1001000011"=> A <= "01111101010101010101110110100110"; -- Index 579 Coefficient 0.95833952996
when "1001000100"=> A <= "011111010100000100110100110100011"; -- Index 580 Coefficient 0.95710487785
when "1001000101"=> A <= "011111010010110010110010001000010"; -- Index 581 Coefficient 0.95585301600
when "1001000110"=> A <= "01111101000101111110011011101000"; -- Index 582 Coefficient 0.95458399155
when "1001000111"=> A <= "01111101000000101101010011111101"; -- Index 583 Coefficient 0.95329785226
when "1001001000"=> A <= "011111001110110101111010111101000"; -- Index 584 Coefficient 0.95199464656
when "1001001001"=> A <= "011111001101011111011001100010011"; -- Index 585 Coefficient 0.95067442352
when "1001001010"=> A <= "011111001100000111110000111101000"; -- Index 586 Coefficient 0.94933723285
when "1001001011"=> A <= "011111001010101111000001011010011"; -- Index 587 Coefficient 0.94798312488
when "1001001100"=> A <= "011111001001010101001011001000010"; -- Index 588 Coefficient 0.94661215060
when "1001001101"=> A <= "011111000111111010001110010100100"; -- Index 589 Coefficient 0.94522436162
when "1001001110"=> A <= "01111100011001111000101100101001"; -- Index 590 Coefficient 0.94381981020
when "1001001111"=> A <= "011111000101000001000010000000010"; -- Index 591 Coefficient 0.94239854922
when "1001010000"=> A <= "011111000011100010110010111100011"; -- Index 592 Coefficient 0.94096063217
when "1001010001"=> A <= "011111000010000011011100011111111"; -- Index 593 Coefficient 0.93950611321
when "1001010010"=> A <= "011111000000100011000100001001101"; -- Index 594 Coefficient 0.93803504710
when "1001010011"=> A <= "011110111111000001100100111000011"; -- Index 595 Coefficient 0.93654748921
when "1001010100"=> A <= "011110111101011111000000101011001"; -- Index 596 Coefficient 0.93504349555
when "1001010101"=> A <= "011110111011111011010111110001001"; -- Index 597 Coefficient 0.93352312276
when "1001010110"=> A <= "011110111010010110101010011001110"; -- Index 598 Coefficient 0.93198642806
when "1001010111"=> A <= "011110111000110000111000110100101"; -- Index 599 Coefficient 0.93043346932
when "1001011000"=> A <= "011110110111001010000011010001010"; -- Index 600 Coefficient 0.92886430500
when "1001011001"=> A <= "011110110101100010001001111111101"; -- Index 601 Coefficient 0.92727899418
when "1001011010"=> A <= "011110110011111001001101001111110"; -- Index 602 Coefficient 0.92567759655
when "1001011011"=> A <= "011110110010001111001101010001110"; -- Index 603 Coefficient 0.92406017240
when "1001011100"=> A <= "011110110000100100001010010110000"; -- Index 604 Coefficient 0.92242678262
when "1001011101"=> A <= "011110101110111000000100101101000"; -- Index 605 Coefficient 0.92077748872
when "1001011110"=> A <= "011110101101001010111100100111100"; -- Index 606 Coefficient 0.91911235278
when "1001011111"=> A <= "011110101011011100110010010110010"; -- Index 607 Coefficient 0.91743143749
when "1001100000"=> A <= "011110101001101101100110001010010"; -- Index 608 Coefficient 0.91573480615
when "1001100001"=> A <= "01111010011111101011000010100101"; -- Index 609 Coefficient 0.91402252263
when "1001100010"=> A <= "011110100110001100001001000110110"; -- Index 610 Coefficient 0.91229465139

```

```

when "1001100011"=> A <= "01110100100011001111000110010000"; -- Index 611 Coefficient 0.91055125750
when "1001100100"=> A <= "01110100010100110100111101000001"; -- Index 612 Coefficient 0.90879240658
when "1001100101"=> A <= "01110100000110010010101111010101"; -- Index 613 Coefficient 0.90701816485
when "1001100110"=> A <= "0111001111011110100001111011110"; -- Index 614 Coefficient 0.90522859913
when "1001100111"=> A <= "01110011101000110110001111101100"; -- Index 615 Coefficient 0.90342377677
when "1001101000"=> A <= "0111001101100111110000010010000"; -- Index 616 Coefficient 0.90160376574
when "1001101001"=> A <= "01110011001010111001111001011110"; -- Index 617 Coefficient 0.89976863455
when "1001101010"=> A <= "01110010111011101111110111101010"; -- Index 618 Coefficient 0.89791845230
when "1001101011"=> A <= "0111001010110001110111111001001"; -- Index 619 Coefficient 0.89605328865
when "1001101100"=> A <= "01110010011101000100010010010011"; -- Index 620 Coefficient 0.89417321381
when "1001101101"=> A <= "01110010001101100010110011100000"; -- Index 621 Coefficient 0.89227829858
when "1001101110"=> A <= "01110001111101111001100101001000"; -- Index 622 Coefficient 0.89036861429
when "1001101111"=> A <= "01110001101110001000101001100110"; -- Index 623 Coefficient 0.88844423284
when "1001110000"=> A <= "01110001011110010000000011010110"; -- Index 624 Coefficient 0.88650522668
when "1001110001"=> A <= "01110001001110001111110100110101"; -- Index 625 Coefficient 0.88455166882
when "1001110010"=> A <= "0111000011111000100000000011111"; -- Index 626 Coefficient 0.88258363281
when "1001110011"=> A <= "0111000010110111100010000110110"; -- Index 627 Coefficient 0.88060119274
when "1001110100"=> A <= "01110000011101100001110000011000"; -- Index 628 Coefficient 0.87860442325
when "1001110101"=> A <= "01110000001101000011011001100111"; -- Index 629 Coefficient 0.87659339952
when "1001110110"=> A <= "01101111111100011101100111000111"; -- Index 630 Coefficient 0.87456819726
when "1001110111"=> A <= "01101111101011110000011011011010"; -- Index 631 Coefficient 0.87252889272
when "1001111000"=> A <= "01101111011010111011111001000101"; -- Index 632 Coefficient 0.87047556268
when "1001111001"=> A <= "0110111100101000000000010101111"; -- Index 633 Coefficient 0.86840828444
when "1001111010"=> A <= "01101110111000111100111010111110"; -- Index 634 Coefficient 0.86632713584
when "1001111011"=> A <= "01101110100111110010100100011011"; -- Index 635 Coefficient 0.86423219522
when "1001111100"=> A <= "01101110010110100001000001110000"; -- Index 636 Coefficient 0.86212354148
when "1001111101"=> A <= "01101110000101001000010101100110"; -- Index 637 Coefficient 0.86000125398
when "1001111110"=> A <= "01101101110011101000100010101010"; -- Index 638 Coefficient 0.85786541264
when "1001111111"=> A <= "01101101100010000001101011101000"; -- Index 639 Coefficient 0.85571609787
when "1010000000"=> A <= "01101101010000010011110011001101"; -- Index 640 Coefficient 0.85355339059
when "1010000001"=> A <= "01101100111110011110111100001001"; -- Index 641 Coefficient 0.85137737223
when "1010000010"=> A <= "01101100101100100011001001001100"; -- Index 642 Coefficient 0.84918812470
when "1010000011"=> A <= "01101100011010100000011101000110"; -- Index 643 Coefficient 0.84698573044
when "1010000100"=> A <= "01101100001000010110111010101010"; -- Index 644 Coefficient 0.84477027237
when "1010000101"=> A <= "01101011110110000110100100101011"; -- Index 645 Coefficient 0.84254183389
when "1010000110"=> A <= "0110101110001110111101110111101"; -- Index 646 Coefficient 0.84030049890
when "1010000111"=> A <= "01101011010001010001101001010101"; -- Index 647 Coefficient 0.83804635179
when "1010001000"=> A <= "0110101011111010110100100101001"; -- Index 648 Coefficient 0.83577947742
when "1010001001"=> A <= "01101010101100000010000001110001"; -- Index 649 Coefficient 0.83349996115
when "1010001010"=> A <= "01101010011001010000010100100101"; -- Index 650 Coefficient 0.83120788880
when "1010001011"=> A <= "01101010000110011000000100111111"; -- Index 651 Coefficient 0.82890334665
when "1010001100"=> A <= "01101001110011011001010101111000"; -- Index 652 Coefficient 0.82658642148
when "1010001101"=> A <= "01101001100000010100001010001100"; -- Index 653 Coefficient 0.82425720051
when "1010001110"=> A <= "01101001001101001000010010011011"; -- Index 654 Coefficient 0.82191577144
when "1010001111"=> A <= "01101000111001110110101000110111"; -- Index 655 Coefficient 0.81956222243
when "1010010000"=> A <= "01101000100110011110011001001010"; -- Index 656 Coefficient 0.81719664208
when "1010010001"=> A <= "01101000010010111111111000101111"; -- Index 657 Coefficient 0.81481911946
when "1010010010"=> A <= "01100111111111011011001010100111"; -- Index 658 Coefficient 0.81242974407
when "1010010011"=> A <= "01100111101011110000010001110010"; -- Index 659 Coefficient 0.81002860588
when "1010010100"=> A <= "0110001110101111111010001010010"; -- Index 660 Coefficient 0.80761579529
when "1010010101"=> A <= "011000111000100001000001100001100"; -- Index 661 Coefficient 0.80519140314
when "1010010110"=> A <= "011000110110000001011000101100010"; -- Index 662 Coefficient 0.80275552070
when "1010010111"=> A <= "01100011001110000100000000011010"; -- Index 663 Coefficient 0.80030823969
when "1010011000"=> A <= "0110001100001111111011111111010"; -- Index 664 Coefficient 0.79784965225
when "1010011001"=> A <= "011000101110011110000000111001000"; -- Index 665 Coefficient 0.79537985093
when "1010011010"=> A <= "011000101011111011011011001001100"; -- Index 666 Coefficient 0.79289892873
when "1010011011"=> A <= "011000101001011000000111001001111"; -- Index 667 Coefficient 0.79040697905
when "1010011100"=> A <= "011000100110110100000101010011010"; -- Index 668 Coefficient 0.78790409571

```

```

when "1010011101"=> A <= "0110010010000111101010111110111"; -- Index 669 Coefficient 0.78539037294
when "1010011110"=> A <= "01100100001101001111001100110010"; -- Index 670 Coefficient 0.78286590539
when "1010011111"=> A <= "0110001111100001110000100010111"; -- Index 671 Coefficient 0.78033078810
when "1010100000"=> A <= "01100011100011100111011001110011"; -- Index 672 Coefficient 0.77778511651
when "1010100001"=> A <= "01100011001110101011010000010100"; -- Index 673 Coefficient 0.77522898647
when "1010100010"=> A <= "01100010111001101001101011001000"; -- Index 674 Coefficient 0.7726249421
when "1010100011"=> A <= "01100010100100100010101101011110"; -- Index 675 Coefficient 0.77008573636
when "1010100100"=> A <= "01100010001111010110011010101000"; -- Index 676 Coefficient 0.76749880994
when "1010100101"=> A <= "01100001111010000100110101110110"; -- Index 677 Coefficient 0.76490181234
when "1010100110"=> A <= "01100001100100101110000010011011"; -- Index 678 Coefficient 0.76229484134
when "1010100111"=> A <= "01100001001111010010000011101000"; -- Index 679 Coefficient 0.75967799508
when "1010101000"=> A <= "01100000111001110000111100110010"; -- Index 680 Coefficient 0.75705137210
when "1010101001"=> A <= "01100000100100001010110001001101"; -- Index 681 Coefficient 0.75441507127
when "1010101010"=> A <= "01100000001110011111100100001111"; -- Index 682 Coefficient 0.75176919186
when "1010101011"=> A <= "01011111111000101111011001001100"; -- Index 683 Coefficient 0.74911383349
when "1010101100"=> A <= "01011111100010111010010011011100"; -- Index 684 Coefficient 0.74644909611
when "1010101101"=> A <= "0101111100110100000010110010110"; -- Index 685 Coefficient 0.74377508007
when "1010101110"=> A <= "01011110110111000001100101010011"; -- Index 686 Coefficient 0.74109188604
when "1010101111"=> A <= "01011110100000111110000011101011"; -- Index 687 Coefficient 0.73839961503
when "1010110000"=> A <= "01011110001010110101110100111000"; -- Index 688 Coefficient 0.73569836841
when "1010110001"=> A <= "01011101110100101000111100010101"; -- Index 689 Coefficient 0.73298824788
when "1010110010"=> A <= "01011101011110010111011101011100"; -- Index 690 Coefficient 0.73026935548
when "1010110011"=> A <= "0101110100100000001011011101001"; -- Index 691 Coefficient 0.72754179356
when "1010110100"=> A <= "01011100110001100110111010011001"; -- Index 692 Coefficient 0.72480566483
when "1010110101"=> A <= "01011100011011000111111101001010"; -- Index 693 Coefficient 0.72206107229
when "1010110110"=> A <= "01011100000100100100100111011000"; -- Index 694 Coefficient 0.71930811927
when "1010110111"=> A <= "01011011101101111100111100100011"; -- Index 695 Coefficient 0.71654690943
when "1010111000"=> A <= "0101101101011101000100000001010"; -- Index 696 Coefficient 0.71377754672
when "1010111001"=> A <= "01011011000000100000110101101100"; -- Index 697 Coefficient 0.71100013540
when "1010111010"=> A <= "01011010101001101100100000101011"; -- Index 698 Coefficient 0.70821478005
when "1010111011"=> A <= "01011010010010110100000100101000"; -- Index 699 Coefficient 0.70542158553
when "1010111100"=> A <= "01011001111011110111100101000100"; -- Index 700 Coefficient 0.70262065700
when "1010111101"=> A <= "01011001100100110111000101100001"; -- Index 701 Coefficient 0.69981209992
when "1010111110"=> A <= "01011001001101110010101001100100"; -- Index 702 Coefficient 0.69699602003
when "1010111111"=> A <= "01011000110110101010010100101111"; -- Index 703 Coefficient 0.69417252335
when "1011000000"=> A <= "01011000011111011110001010100111"; -- Index 704 Coefficient 0.69134171618
when "1011000001"=> A <= "01011000001000001110001110110000"; -- Index 705 Coefficient 0.68850370511
when "1011000010"=> A <= "01010111110000111010100100110001"; -- Index 706 Coefficient 0.68565859698
when "1011000011"=> A <= "01010111011001100011010000001111"; -- Index 707 Coefficient 0.68280649890
when "1011000100"=> A <= "01010111000010001000010100110001"; -- Index 708 Coefficient 0.67994751827
when "1011000101"=> A <= "01010110101010101001110101111110"; -- Index 709 Coefficient 0.67708176271
when "1011000110"=> A <= "01010110010011000111110111011101"; -- Index 710 Coefficient 0.67420934012
when "1011000111"=> A <= "01010101111011100010011100111000"; -- Index 711 Coefficient 0.67133035866
when "1011001000"=> A <= "01010101100011111001101001110110"; -- Index 712 Coefficient 0.66844492670
when "1011001001"=> A <= "01010101001100001101100010000001"; -- Index 713 Coefficient 0.66555315288
when "1011001010"=> A <= "01010100110100011110001001000010"; -- Index 714 Coefficient 0.66265514608
when "1011001011"=> A <= "01010100011100101011100010100101"; -- Index 715 Coefficient 0.65975101541
when "1011001100"=> A <= "01010100000100110101110010010100"; -- Index 716 Coefficient 0.65684087020
when "1011001101"=> A <= "01010011101100111100111011111010"; -- Index 717 Coefficient 0.65392482002
when "1011001110"=> A <= "01010011010101000001000011000011"; -- Index 718 Coefficient 0.65100297466
when "1011001111"=> A <= "01010010111101000010001011011011"; -- Index 719 Coefficient 0.64807544412
when "1011010000"=> A <= "01010010100101000000011000101111"; -- Index 720 Coefficient 0.64514233863
when "1011010001"=> A <= "01010010001100111011101110101100"; -- Index 721 Coefficient 0.64220376861
when "1011010010"=> A <= "01010001110100110100010000111111"; -- Index 722 Coefficient 0.63925984469
when "1011010011"=> A <= "0101000101110010101000011010111"; -- Index 723 Coefficient 0.63631067772
when "1011010100"=> A <= "01010001000100011101001001100011"; -- Index 724 Coefficient 0.63335637874
when "1011010101"=> A <= "010100000101100001101100111010000"; -- Index 725 Coefficient 0.63039705896
when "1011010110"=> A <= "01010000010011111011100000011110"; -- Index 726 Coefficient 0.62743282980

```

```

when "1011010111"=> A <= "0100111111011100110111000001101"; -- Index 727 Coefficient 0.62446380287
when "1011011000"=> A <= "01001111100011001111110010111110"; -- Index 728 Coefficient 0.62149008995
when "1011011001"=> A <= "0100111100101010110010100001111"; -- Index 729 Coefficient 0.61851180300
when "1011011010"=> A <= "01001110011001001101001111110011"; -- Index 730 Coefficient 0.61552905414
when "1011011011"=> A <= "01001110011001111100011001011010"; -- Index 731 Coefficient 0.61254195568
when "1011011100"=> A <= "0100111000001011100000100110101"; -- Index 732 Coefficient 0.60955062008
when "1011011101"=> A <= "01001101101000111001100101111000"; -- Index 733 Coefficient 0.60655515996
when "1011011110"=> A <= "0100110101000001010100000010011"; -- Index 734 Coefficient 0.60355568810
when "1011011111"=> A <= "01001100110111101110010111111001"; -- Index 735 Coefficient 0.60055231742
when "1011100000"=> A <= "01001100011111000101110000011110"; -- Index 736 Coefficient 0.59754516101
when "1011100001"=> A <= "01001100000110011011001101110100"; -- Index 737 Coefficient 0.59453433207
when "1011100010"=> A <= "0100101110110110111011001101111"; -- Index 738 Coefficient 0.59151994398
when "1011100011"=> A <= "01001011010101000000100110000010"; -- Index 739 Coefficient 0.58850211021
when "1011100100"=> A <= "01001010111100010000101000100010"; -- Index 740 Coefficient 0.58548094438
when "1011100101"=> A <= "0100101010001101111011111000011"; -- Index 741 Coefficient 0.58245656024
when "1011100110"=> A <= "01001010001010101011101101011001"; -- Index 742 Coefficient 0.57942907167
when "1011100111"=> A <= "0100100111000111011011011011000"; -- Index 743 Coefficient 0.57639859263
when "1011101000"=> A <= "01001001011001000000100000110111"; -- Index 744 Coefficient 0.57336523723
when "1011101001"=> A <= "01001001000000001000101101101010"; -- Index 745 Coefficient 0.57032911967
when "1011101010"=> A <= "01001000100111001111100001100111"; -- Index 746 Coefficient 0.56729035425
when "1011101011"=> A <= "01001000001110010101000000100100"; -- Index 747 Coefficient 0.56424905540
when "1011101100"=> A <= "0100011111010101001001110010110"; -- Index 748 Coefficient 0.56120533760
when "1011101101"=> A <= "01000111011100011100001110110011"; -- Index 749 Coefficient 0.55815931546
when "1011101110"=> A <= "01000111000011011110000101110010"; -- Index 750 Coefficient 0.55511110365
when "1011101111"=> A <= "01000110101010011110110111001001"; -- Index 751 Coefficient 0.55206081694
when "1011110000"=> A <= "01000110010001011110100110101111"; -- Index 752 Coefficient 0.54900857016
when "1011110001"=> A <= "01000101111000011101011000011011"; -- Index 753 Coefficient 0.54595447825
when "1011110010"=> A <= "01000101011111011011010000000011"; -- Index 754 Coefficient 0.54289865617
when "1011110011"=> A <= "01000101000110011000010001011110"; -- Index 755 Coefficient 0.53984121899
when "1011110100"=> A <= "01000100101101010100100000100101"; -- Index 756 Coefficient 0.53678228180
when "1011110101"=> A <= "01000100010100010000000001001101"; -- Index 757 Coefficient 0.53372195978
when "1011110110"=> A <= "01000011111011001010110111001111"; -- Index 758 Coefficient 0.53066036815
when "1011110111"=> A <= "01000011100010000101000110100010"; -- Index 759 Coefficient 0.52759762217
when "1011111000"=> A <= "01000011001000111110110010111110"; -- Index 760 Coefficient 0.52453383716
when "1011111001"=> A <= "010000101011111100000000011010"; -- Index 761 Coefficient 0.52146912847
when "1011111010"=> A <= "01000010010110110000110010101111"; -- Index 762 Coefficient 0.51840361147
when "1011111011"=> A <= "01000001111101101001001101110011"; -- Index 763 Coefficient 0.51533740159
when "1011111100"=> A <= "01000001100100100001010101011111"; -- Index 764 Coefficient 0.51227061426
when "1011111101"=> A <= "01000001001011011001001101101100"; -- Index 765 Coefficient 0.50920336495
when "1011111110"=> A <= "01000000110010010000111010010000"; -- Index 766 Coefficient 0.50613576914
when "1011111111"=> A <= "01000000011001001000011111000100"; -- Index 767 Coefficient 0.50306794232
when "1100000000"=> A <= "01000000000000000000000000000000"; -- Index 768 Coefficient 0.50000000000
when "1100000001"=> A <= "0011111110011011011110000011100"; -- Index 769 Coefficient 0.49693205768
when "1100000010"=> A <= "00111111001101101111000101110000"; -- Index 770 Coefficient 0.49386423086
when "1100000011"=> A <= "00111110110100100110110010010100"; -- Index 771 Coefficient 0.49079663505
when "1100000100"=> A <= "00111110011011011110101010100001"; -- Index 772 Coefficient 0.48772938574
when "1100000101"=> A <= "00111110000010010110110010001101"; -- Index 773 Coefficient 0.48466259841
when "1100000110"=> A <= "00111101101001001111001101010001"; -- Index 774 Coefficient 0.48159638853
when "1100000111"=> A <= "0011110101000000011111111100110"; -- Index 775 Coefficient 0.47853087153
when "1100001000"=> A <= "0011110011011100000100101000010"; -- Index 776 Coefficient 0.47546616284
when "1100001001"=> A <= "00111100011101111010111001011110"; -- Index 777 Coefficient 0.47240237783
when "1100001010"=> A <= "00111100000100110101001000110001"; -- Index 778 Coefficient 0.46933963185
when "1100001011"=> A <= "001111011101011101111111110110011"; -- Index 779 Coefficient 0.46627804022
when "1100001100"=> A <= "00111101101001010101011111011011"; -- Index 780 Coefficient 0.46321771820
when "1100001101"=> A <= "00111101011100110011011110100010"; -- Index 781 Coefficient 0.46015878101
when "1100001110"=> A <= "0011110101000001001001011111111101"; -- Index 782 Coefficient 0.45710134383
when "1100001111"=> A <= "001111010000111100010100111100101"; -- Index 783 Coefficient 0.45404552175
when "1100010000"=> A <= "001111001101110100001011001010001"; -- Index 784 Coefficient 0.45099142984

```

```

when "1100010001"=> A <= "00111001010101100001001000110111"; -- Index 785 Coefficient 0.44793918306
when "1100010010"=> A <= "00111000111100100001111010001110"; -- Index 786 Coefficient 0.44488889635
when "1100010011"=> A <= "00111000100011100011110001001101"; -- Index 787 Coefficient 0.44184068454
when "1100010100"=> A <= "00111000001010100110110001101010"; -- Index 788 Coefficient 0.43879466240
when "1100010101"=> A <= "00110111110001101010111111011100"; -- Index 789 Coefficient 0.43575094460
when "1100010110"=> A <= "00110111011000110000011110011001"; -- Index 790 Coefficient 0.43270964575
when "1100010111"=> A <= "0011011011111111011010010010110"; -- Index 791 Coefficient 0.42967088033
when "1100011000"=> A <= "00110110100110111111011111001001"; -- Index 792 Coefficient 0.42663476277
when "1100011001"=> A <= "00110110001110001001001000101000"; -- Index 793 Coefficient 0.42360140737
when "1100011010"=> A <= "00110101110101010100010010100111"; -- Index 794 Coefficient 0.42057092833
when "1100011011"=> A <= "00110101011100100001000000111101"; -- Index 795 Coefficient 0.41754343976
when "1100011100"=> A <= "00110101000011101111010111011110"; -- Index 796 Coefficient 0.41451905562
when "1100011101"=> A <= "00110100101010111111011001111110"; -- Index 797 Coefficient 0.41149788979
when "1100011110"=> A <= "00110100010010010001001100010001"; -- Index 798 Coefficient 0.40848005602
when "1100011111"=> A <= "00110011111001100100110010001100"; -- Index 799 Coefficient 0.40546566793
when "1100100000"=> A <= "00110011100000111010001111100010"; -- Index 800 Coefficient 0.40245483899
when "1100100001"=> A <= "00110011001000010001101000000111"; -- Index 801 Coefficient 0.39944768258
when "1100100010"=> A <= "00110010101111101010111111101101"; -- Index 802 Coefficient 0.39644431190
when "1100100011"=> A <= "00110010010111000110011010001000"; -- Index 803 Coefficient 0.39344484004
when "1100100100"=> A <= "00110001111110100011111011001011"; -- Index 804 Coefficient 0.39044937992
when "1100100101"=> A <= "00110001100110000011100110100110"; -- Index 805 Coefficient 0.38745804432
when "1100100110"=> A <= "0011000100110110010110000001101"; -- Index 806 Coefficient 0.38447094586
when "1100100111"=> A <= "00110000110101001001101011110001"; -- Index 807 Coefficient 0.38148819700
when "1100101000"=> A <= "00110000011100110000001101000010"; -- Index 808 Coefficient 0.37850991005
when "1100101001"=> A <= "00110000000100011001000111110011"; -- Index 809 Coefficient 0.37553619713
when "1100101010"=> A <= "00101111101100000100011111110010"; -- Index 810 Coefficient 0.37256717020
when "1100101011"=> A <= "00101111010011110010011000110000"; -- Index 811 Coefficient 0.36960294104
when "1100101100"=> A <= "0010111011101100010110110011101"; -- Index 812 Coefficient 0.36664362126
when "1100101101"=> A <= "00101110100011010101111100101001"; -- Index 813 Coefficient 0.36368932228
when "1100101110"=> A <= "00101110001011001011101111000001"; -- Index 814 Coefficient 0.36074015531
when "1100101111"=> A <= "00101101110011000100010001010100"; -- Index 815 Coefficient 0.35779623139
when "1100110000"=> A <= "0010110101101011111100111010001"; -- Index 816 Coefficient 0.35485766137
when "1100110001"=> A <= "00101101000010111101110100100101"; -- Index 817 Coefficient 0.35192455588
when "1100110010"=> A <= "00101100101010111110111100111101"; -- Index 818 Coefficient 0.34899702534
when "1100110011"=> A <= "00101100010011000011000100000110"; -- Index 819 Coefficient 0.34607517998
when "1100110100"=> A <= "00101011111011001010001101101100"; -- Index 820 Coefficient 0.34315912980
when "1100110101"=> A <= "00101011100011010100011101011011"; -- Index 821 Coefficient 0.34024898459
when "1100110110"=> A <= "00101011001011100001110110111110"; -- Index 822 Coefficient 0.33734485392
when "1100110111"=> A <= "00101010110011110010011101111111"; -- Index 823 Coefficient 0.33444684712
when "1100111000"=> A <= "00101010011100000110010110001010"; -- Index 824 Coefficient 0.33155507330
when "1100111001"=> A <= "00101010000100011101100011001000"; -- Index 825 Coefficient 0.32866964134
when "1100111010"=> A <= "00101001101100111000001000100011"; -- Index 826 Coefficient 0.32579065988
when "1100111011"=> A <= "00101001010101010110001010000010"; -- Index 827 Coefficient 0.32291823729
when "1100111100"=> A <= "00101000111101110111101011001111"; -- Index 828 Coefficient 0.32005248173
when "1100111101"=> A <= "00101000100110011100101111110001"; -- Index 829 Coefficient 0.31719350110
when "1100111110"=> A <= "00101000001111000101011011001111"; -- Index 830 Coefficient 0.31434140302
when "1100111111"=> A <= "00100111110111110001110001010000"; -- Index 831 Coefficient 0.31149629489
when "1101000000"=> A <= "00100111100000100001110101011001"; -- Index 832 Coefficient 0.30865828382
when "1101000001"=> A <= "00100111001001010101010101010001"; -- Index 833 Coefficient 0.30582747665
when "1101000010"=> A <= "00100110110010001101010110011100"; -- Index 834 Coefficient 0.30300397997
when "1101000011"=> A <= "00100110011011001000111010011111"; -- Index 835 Coefficient 0.30018790008
when "1101000100"=> A <= "00100110000100001000011010111100"; -- Index 836 Coefficient 0.29737934300
when "1101000101"=> A <= "00100101101101001011111011011000"; -- Index 837 Coefficient 0.29457841447
when "1101000110"=> A <= "0010010101011001001011111010101"; -- Index 838 Coefficient 0.29178521995
when "1101000111"=> A <= "00100100111111011111001010010100"; -- Index 839 Coefficient 0.28899986460
when "1101001000"=> A <= "00100100101000010111011111110110"; -- Index 840 Coefficient 0.28622245328
when "1101001001"=> A <= "00100100010010000011000011011101"; -- Index 841 Coefficient 0.28345309057
when "1101001010"=> A <= "00100011111011011011011000101000"; -- Index 842 Coefficient 0.28069188073

```

```

when "1101001011"=> A <= "00100011100100111000000010110110"; -- Index 843 Coefficient 0.27793892771
when "1101001100"=> A <= "00100011001110011001000101100111"; -- Index 844 Coefficient 0.27519433517
when "1101001101"=> A <= "00100010110111111110100100010111"; -- Index 845 Coefficient 0.27245820644
when "1101001110"=> A <= "00100010100001101000100010100100"; -- Index 846 Coefficient 0.26973064452
when "1101001111"=> A <= "00100010001011010111000011101011"; -- Index 847 Coefficient 0.26701175212
when "1101010000"=> A <= "00100001110101001010001011001000"; -- Index 848 Coefficient 0.26430163159
when "1101010001"=> A <= "00100001011111000001111100010101"; -- Index 849 Coefficient 0.26160038497
when "1101010010"=> A <= "00100001001000111110011010101101"; -- Index 850 Coefficient 0.25890811396
when "1101010011"=> A <= "0010000011001011111101001101010"; -- Index 851 Coefficient 0.25622491993
when "1101010100"=> A <= "00100000011101000101101100100100"; -- Index 852 Coefficient 0.25355090389
when "1101010101"=> A <= "00100000000111010000100110110100"; -- Index 853 Coefficient 0.25088616651
when "1101010110"=> A <= "00011111110001100000011011110001"; -- Index 854 Coefficient 0.24823080814
when "1101010111"=> A <= "00011111011011110101001110110011"; -- Index 855 Coefficient 0.24558492873
when "1101011000"=> A <= "00011111000110001111000011001110"; -- Index 856 Coefficient 0.24294862790
when "1101011001"=> A <= "00011110110000101101111100011000"; -- Index 857 Coefficient 0.24032200492
when "1101011010"=> A <= "00011110011011010001111101100101"; -- Index 858 Coefficient 0.23770515866
when "1101011011"=> A <= "0001111000010111011001010001010"; -- Index 859 Coefficient 0.23509818766
when "1101011100"=> A <= "000111011100001010011000101011000"; -- Index 860 Coefficient 0.23250119006
when "1101011101"=> A <= "00011101011011011101010010100010"; -- Index 861 Coefficient 0.22991426364
when "1101011110"=> A <= "00011101000110010110010100111000"; -- Index 862 Coefficient 0.22733750579
when "1101011111"=> A <= "00011100110001010100101111101100"; -- Index 863 Coefficient 0.22477101353
when "1101100000"=> A <= "00011100011100011000100110001101"; -- Index 864 Coefficient 0.22221488349
when "1101100001"=> A <= "0001110000011100001111011101001"; -- Index 865 Coefficient 0.21966921190
when "1101100010"=> A <= "00011011110010110000110011001110"; -- Index 866 Coefficient 0.21713409461
when "1101100011"=> A <= "00011011011110000101010000001001"; -- Index 867 Coefficient 0.21460962706
when "1101100100"=> A <= "00011011001001011111010101100110"; -- Index 868 Coefficient 0.21209590429
when "1101100101"=> A <= "00011010110100111111000110110001"; -- Index 869 Coefficient 0.20959302095
when "1101100110"=> A <= "00011010100000100100100110110100"; -- Index 870 Coefficient 0.20710107127
when "1101100111"=> A <= "00011010001100001111111000111000"; -- Index 871 Coefficient 0.20462014907
when "1101101000"=> A <= "00011001111000000001000000000110"; -- Index 872 Coefficient 0.20215034775
when "1101101001"=> A <= "00011001100011110111111111100110"; -- Index 873 Coefficient 0.19969176031
when "1101101010"=> A <= "00011001001111110100111010011110"; -- Index 874 Coefficient 0.19724447930
when "1101101011"=> A <= "00011000111011110111110011110100"; -- Index 875 Coefficient 0.19480859686
when "1101101100"=> A <= "00011000101000000000101110101110"; -- Index 876 Coefficient 0.19238420471
when "1101101101"=> A <= "00011000010100001111101110001110"; -- Index 877 Coefficient 0.18997139412
when "1101101110"=> A <= "0001100000000100100110101011001"; -- Index 878 Coefficient 0.18757025593
when "1101101111"=> A <= "00010111101101000000000111010001"; -- Index 879 Coefficient 0.18518088054
when "1101110000"=> A <= "00010111011001100001100110110110"; -- Index 880 Coefficient 0.18280335792
when "1101110001"=> A <= "00010111000110001001010111001001"; -- Index 881 Coefficient 0.18043777757
when "1101110010"=> A <= "00010110110010110111011011001001"; -- Index 882 Coefficient 0.17808422856
when "1101110011"=> A <= "00010110011111101011110101110100"; -- Index 883 Coefficient 0.17574279949
when "1101110100"=> A <= "00010110001100100110101010001000"; -- Index 884 Coefficient 0.17341357852
when "1101110101"=> A <= "00010101111001100111111011000001"; -- Index 885 Coefficient 0.17109665335
when "1101110110"=> A <= "00010101100110101111101011011011"; -- Index 886 Coefficient 0.16879211120
when "1101110111"=> A <= "00010101010011111101111110001111"; -- Index 887 Coefficient 0.16650003885
when "1101111000"=> A <= "00010101000001010010110110010111"; -- Index 888 Coefficient 0.16422052258
when "1101111001"=> A <= "00010100101110101110010110101011"; -- Index 889 Coefficient 0.16195364821
when "1101111010"=> A <= "00010100011100010000100010000011"; -- Index 890 Coefficient 0.15969950110
when "1101111011"=> A <= "00010100001001111001011011010101"; -- Index 891 Coefficient 0.15745816611
when "1101111100"=> A <= "00010011110111101001000101010110"; -- Index 892 Coefficient 0.15522972763
when "1101111101"=> A <= "00010011100101011111100010111010"; -- Index 893 Coefficient 0.15301426956
when "1101111110"=> A <= "00010011010011011100110110110100"; -- Index 894 Coefficient 0.15081187530
when "1101111111"=> A <= "00010011000001100001000011110111"; -- Index 895 Coefficient 0.14862262777
when "1110000000"=> A <= "00010010101111101100001100110011"; -- Index 896 Coefficient 0.14644660941
when "1110000001"=> A <= "0001001001110111110010100011000"; -- Index 897 Coefficient 0.14428390213
when "1110000010"=> A <= "00010010001100010111011101010110"; -- Index 898 Coefficient 0.14213458736
when "1110000011"=> A <= "00010001111010110111101010011010"; -- Index 899 Coefficient 0.13999874602
when "1110000100"=> A <= "00010001101001011110111110010000"; -- Index 900 Coefficient 0.13787645852

```



```

when "1110000101"=> A <= "00010001011000001101011011100101"; -- Index 901 Coefficient 0.13576780478
when "1110000110"=> A <= "00010001000111000011000101000010"; -- Index 902 Coefficient 0.13367286416
when "1110000111"=> A <= "0001000011010111111111101010001"; -- Index 903 Coefficient 0.13159171556
when "1110001000"=> A <= "00010000100101000100000110111011"; -- Index 904 Coefficient 0.12952443732
when "1110001001"=> A <= "0001000001010000111100100100110"; -- Index 905 Coefficient 0.12747110728
when "1110001010"=> A <= "0001000000011100010011000111001"; -- Index 906 Coefficient 0.12543180274
when "1110001011"=> A <= "00001111110010111100100110011001"; -- Index 907 Coefficient 0.12340660048
when "1110001100"=> A <= "00001111100010011110001111101000"; -- Index 908 Coefficient 0.12139557675
when "1110001101"=> A <= "00001111010010000111010111001010"; -- Index 909 Coefficient 0.11939880726
when "1110001110"=> A <= "000011110000011011111111100001"; -- Index 910 Coefficient 0.11741636719
when "1110001111"=> A <= "00001110110001110000001011001011"; -- Index 911 Coefficient 0.11544833118
when "1110010000"=> A <= "0000111010000110111111100101010"; -- Index 912 Coefficient 0.11349477332
when "1110010001"=> A <= "00001110010001110111010110011010"; -- Index 913 Coefficient 0.11155576716
when "1110010010"=> A <= "00001110000010000110011010111000"; -- Index 914 Coefficient 0.10963138571
when "1110010011"=> A <= "00001101110010011101001100100000"; -- Index 915 Coefficient 0.10772170142
when "1110010100"=> A <= "00001101100010111011101101101101"; -- Index 916 Coefficient 0.10582678619
when "1110010101"=> A <= "0000110101001110001000000010111"; -- Index 917 Coefficient 0.103946711135
when "1110010110"=> A <= "00001101000100010000001000010110"; -- Index 918 Coefficient 0.10208154770
when "1110010111"=> A <= "00001100110101000110000110100010"; -- Index 919 Coefficient 0.10023136545
when "1110011000"=> A <= "0000110010011000001111101110000"; -- Index 920 Coefficient 0.09839623426
when "1110011001"=> A <= "00001100010111001001110000010100"; -- Index 921 Coefficient 0.09657622323
when "1110011010"=> A <= "00001100001000010111100000100010"; -- Index 922 Coefficient 0.09477140087
when "1110011011"=> A <= "00001011111001101101010000101011"; -- Index 923 Coefficient 0.09298183515
when "1110011100"=> A <= "0000101110101100101100001011111"; -- Index 924 Coefficient 0.09120759342
when "1110011101"=> A <= "00001011011100110000111001110000"; -- Index 925 Coefficient 0.08944874250
when "1110011110"=> A <= "00001011001110011110110111001010"; -- Index 926 Coefficient 0.08770534861
when "1110011111"=> A <= "00001011000000010100111101011011"; -- Index 927 Coefficient 0.08597747737
when "1110100000"=> A <= "00001010110010010011001110101110"; -- Index 928 Coefficient 0.08426519385
when "1110100001"=> A <= "00001010100100011001101101001110"; -- Index 929 Coefficient 0.08256856251
when "1110100010"=> A <= "00001010010110101000011011000100"; -- Index 930 Coefficient 0.08088764722
when "1110100011"=> A <= "0000101000100011111011010011000"; -- Index 931 Coefficient 0.07922251128
when "1110100100"=> A <= "0000100111101101111010101010000"; -- Index 932 Coefficient 0.07757321738
when "1110100101"=> A <= "00001001101110000110010101110010"; -- Index 933 Coefficient 0.07593982760
when "1110100110"=> A <= "00001001100000110110010110000010"; -- Index 934 Coefficient 0.07432240345
when "1110100111"=> A <= "00001001010011101110110000000011"; -- Index 935 Coefficient 0.07272100582
when "1110101000"=> A <= "0000100100011010111100101110110"; -- Index 936 Coefficient 0.07113569500
when "1110101001"=> A <= "00001000111001111000111001011011"; -- Index 937 Coefficient 0.06956653068
when "1110101010"=> A <= "0000100010110100101010100110010"; -- Index 938 Coefficient 0.06801357194
when "1110101011"=> A <= "00001000100000100101000001110111"; -- Index 939 Coefficient 0.06647687724
when "1110101100"=> A <= "00001000010100000111111010100111"; -- Index 940 Coefficient 0.06495650445
when "1110101101"=> A <= "00001000000111110011011000111101"; -- Index 941 Coefficient 0.06345251079
when "1110101110"=> A <= "00000111111011100111011110110011"; -- Index 942 Coefficient 0.06196495290
when "1110101111"=> A <= "00000111101111100100001110000001"; -- Index 943 Coefficient 0.06049388679
when "1110110000"=> A <= "00000111100011101001101000011101"; -- Index 944 Coefficient 0.05903936783
when "1110110001"=> A <= "000001110101111101111011111110"; -- Index 945 Coefficient 0.05760145078
when "1110110010"=> A <= "00000111001100001110100110010111"; -- Index 946 Coefficient 0.05618018980
when "1110110011"=> A <= "0000011100000101110001101011100"; -- Index 947 Coefficient 0.05477563838
when "1110110100"=> A <= "00000110110101010110100110111110"; -- Index 948 Coefficient 0.05338784940
when "1110110101"=> A <= "00000110101010000111110100101101"; -- Index 949 Coefficient 0.05201687512
when "1110110110"=> A <= "00000110011111000001111000011000"; -- Index 950 Coefficient 0.05066276715
when "1110110111"=> A <= "00000110010100000100110011101101"; -- Index 951 Coefficient 0.04932557648
when "1110111000"=> A <= "000001100010010100000101000011000"; -- Index 952 Coefficient 0.04800535344
when "1110111001"=> A <= "00000101111110100101011000000011"; -- Index 953 Coefficient 0.04670214774
when "1110111010"=> A <= "00000101110100000011000100011000"; -- Index 954 Coefficient 0.04541600845
when "1110111011"=> A <= "00000101101001101001101110111110"; -- Index 955 Coefficient 0.04414698400
when "1110111100"=> A <= "00000101011111011001011001011101"; -- Index 956 Coefficient 0.04289512215
when "1110111101"=> A <= "00000101010101010010000101011010"; -- Index 957 Coefficient 0.04166047004
when "1110111110"=> A <= "00000101001011010011110100011000"; -- Index 958 Coefficient 0.04044307415

```

```

when "1110111111"=> A <= "00000101000001011110100111110111"; -- Index 959 Coefficient 0.03924298033
when "1111000000"=> A <= "00000100110111110010100001100010"; -- Index 960 Coefficient 0.03806023374
when "1111000001"=> A <= "0000010010111000111100010101101"; -- Index 961 Coefficient 0.03689487893
when "1111000010"=> A <= "00000100100100110101101100111100"; -- Index 962 Coefficient 0.03574695976
when "1111000011"=> A <= "00000100011011100101000001101001"; -- Index 963 Coefficient 0.03461651946
when "1111000100"=> A <= "00000100010010011101100010010010"; -- Index 964 Coefficient 0.03350360058
when "1111000101"=> A <= "00000100001001011111010000010000"; -- Index 965 Coefficient 0.03240824503
when "1111000110"=> A <= "00000100000000101010001100111100"; -- Index 966 Coefficient 0.03133049404
when "1111000111"=> A <= "0000001111011111110011001101100"; -- Index 967 Coefficient 0.03027038820
when "1111001000"=> A <= "00000011101111011011110111110110"; -- Index 968 Coefficient 0.02922796741
when "1111001001"=> A <= "00000011100111000010101000101111"; -- Index 969 Coefficient 0.02820327092
when "1111001010"=> A <= "00000011011110110010101010101010"; -- Index 970 Coefficient 0.02719633731
when "1111001011"=> A <= "00000011010110101100000111110111"; -- Index 971 Coefficient 0.02620720449
when "1111001100"=> A <= "00000011001110101110111000100111"; -- Index 972 Coefficient 0.02523590970
when "1111001101"=> A <= "00000011000110111011000001001001"; -- Index 973 Coefficient 0.02428248952
when "1111001110"=> A <= "0000001011111010000100010101001"; -- Index 974 Coefficient 0.02334697982
when "1111001111"=> A <= "00000010110111101111011110010100"; -- Index 975 Coefficient 0.02242941585
when "1111010000"=> A <= "00000010110000010111110101010010"; -- Index 976 Coefficient 0.02152983213
when "1111010001"=> A <= "00000010101001001001101000101110"; -- Index 977 Coefficient 0.02064826255
when "1111010010"=> A <= "00000010100010000100111001101110"; -- Index 978 Coefficient 0.01978474029
when "1111010011"=> A <= "00000010011011001001101001011000"; -- Index 979 Coefficient 0.01893929787
when "1111010100"=> A <= "0000001001010001011111000110001"; -- Index 980 Coefficient 0.01811196710
when "1111010101"=> A <= "00000010001101101111101000111011"; -- Index 981 Coefficient 0.01730277915
when "1111010110"=> A <= "00000010000111010000111010111000"; -- Index 982 Coefficient 0.01651176448
when "1111010111"=> A <= "00000010000000111011101111101000"; -- Index 983 Coefficient 0.01573895286
when "1111011000"=> A <= "00000001111010110000001000001001"; -- Index 984 Coefficient 0.01498437340
when "1111011001"=> A <= "00000001110100101110000101011000"; -- Index 985 Coefficient 0.01424805451
when "1111011010"=> A <= "00000001101110110101101000010001"; -- Index 986 Coefficient 0.01353002390
when "1111011011"=> A <= "00000001101001000110110001101110"; -- Index 987 Coefficient 0.01283030861
when "1111011100"=> A <= "00000001100011100001100010100111"; -- Index 988 Coefficient 0.01214893498
when "1111011101"=> A <= "00000001011110000101111011110100"; -- Index 989 Coefficient 0.01148592867
when "1111011110"=> A <= "00000001011000110011111110001010"; -- Index 990 Coefficient 0.01084131464
when "1111011111"=> A <= "00000001010011101011101010011101"; -- Index 991 Coefficient 0.01021511716
when "1111100000"=> A <= "00000001001110101101000001100000"; -- Index 992 Coefficient 0.00960735980
when "1111100001"=> A <= "00000001001001111000000100000100"; -- Index 993 Coefficient 0.00901806545
when "1111100010"=> A <= "00000001000101001100110010111001"; -- Index 994 Coefficient 0.00844725628
when "1111100011"=> A <= "00000001000000101011001110101100"; -- Index 995 Coefficient 0.00789495381
when "1111100100"=> A <= "00000000111100010011011000001011"; -- Index 996 Coefficient 0.00736117881
when "1111100101"=> A <= "00000000111000000101010000000001"; -- Index 997 Coefficient 0.00684595138
when "1111100110"=> A <= "00000000110100000000110110110110"; -- Index 998 Coefficient 0.00634929092
when "1111100111"=> A <= "00000000110000000110001101010101"; -- Index 999 Coefficient 0.00587121613
when "1111101000"=> A <= "00000000101100010101010100000010"; -- Index 1000 Coefficient 0.00541174502
when "1111101001"=> A <= "00000000101000101110001011100011"; -- Index 1001 Coefficient 0.00497089487
when "1111101010"=> A <= "0000000010010101010000110100011101"; -- Index 1002 Coefficient 0.00454868229
when "1111101011"=> A <= "00000000100001111101001111010000"; -- Index 1003 Coefficient 0.00414512317
when "1111101100"=> A <= "00000000011110110011011100011110"; -- Index 1004 Coefficient 0.00376023270
when "1111101101"=> A <= "0000000001101110011011100100110"; -- Index 1005 Coefficient 0.00339402538
when "1111101110"=> A <= "00000000011000111101010000000101"; -- Index 1006 Coefficient 0.00304651500
when "1111101111"=> A <= "00000000010110010000110111011000"; -- Index 1007 Coefficient 0.00271771463
when "1111110000"=> A <= "00000000010011101110010010111000"; -- Index 1008 Coefficient 0.00240763666
when "1111110001"=> A <= "00000000010001010101100011000000"; -- Index 1009 Coefficient 0.00211629277
when "1111110010"=> A <= "00000000001111000110101000000111"; -- Index 1010 Coefficient 0.00184369391
when "1111110011"=> A <= "00000000001101000001100010100010"; -- Index 1011 Coefficient 0.00158985035
when "1111110100"=> A <= "00000000001011000110010010100110"; -- Index 1012 Coefficient 0.00135477166
when "1111110101"=> A <= "00000000001001010100111000100111"; -- Index 1013 Coefficient 0.00113846668
when "1111110110"=> A <= "000000000001111011010101000110101"; -- Index 1014 Coefficient 0.00094094355
when "1111110111"=> A <= "00000000000110001111100111100001"; -- Index 1015 Coefficient 0.00076220971
when "1111111000"=> A <= "00000000000100111011110000111001"; -- Index 1016 Coefficient 0.00060227190

```

```

when "111111001"=> A <= "00000000000011110001110001001010"; -- Index 1017 Coefficient 0.00046113612
when "111111010"=> A <= "00000000000010110001101000100000"; -- Index 1018 Coefficient 0.00033880771
when "111111011"=> A <= "00000000000001111011010111000100"; -- Index 1019 Coefficient 0.00023529125
when "111111100"=> A <= "00000000000001001110111100111111"; -- Index 1020 Coefficient 0.00015059065
when "111111101"=> A <= "00000000000001011000110100101111"; -- Index 1021 Coefficient 0.00008470910
when "111111110"=> A <= "0000000000000010011101111010011"; -- Index 1022 Coefficient 0.00003764908
when "111111111"=> A <= "00000000000000000100111011110101"; -- Index 1023 Coefficient 0.00000941236
when others      => A <= "00000000000000000000000000000000"; -- Index para casos irrelevantes
end case;
end process;
end LUTable;

```