



Universidad Autónoma de Querétaro

Facultad de Ingeniería

**Sistema basado en FPGA para detección de Sag y Swell en
Fresadora CNC mediante Micro Algoritmos Genéticos y
análisis de su dinámica ante el disturbio**

Tesis

**Que como parte de los requisitos para obtener el título de
Ingeniero Electromecánico**

Presenta:

Artvin Darién González Abreu

Dirigido por:

Dr. Roque Alfredo Osornio Ríos

Codirigido por:

Dr. Arturo Yosimar Jaen Cuellar

San Juan del Río, Querétaro, Agosto de 2016



UNIVERSIDAD AUTÓNOMA DE QUERÉTARO
FACULTAD DE INGENIERÍA
CAMPUS SAN JUAN DEL RÍO



**SISTEMA BASADO EN FPGA PARA DETECCIÓN DE SAG Y SWELL EN FRESADORA
CNC MEDIANTE MICRO ALGORITMOS GENÉTICOS Y ANÁLISIS DE SU DINÁMICA
ANTE EL DISTURBIO**

TESIS

Como parte de los requisitos para obtener el título de

INGENIERO ELECTROMECAÁNICO

Línea Terminal

DISEÑO Y MANUFACTURA

Presenta

ARTVIN DARIÉN GONZÁLEZ ABREU

Dirigido por:

DR. ROQUE ALFREDO OSORNIO RIOS

SINODALES:

DR. ROQUE ALFREDO OSORNIO RÍOS (5620)
Presidente

DR. ARTURO YOSIMAR JAEN CUELLAR (11825)
Secretario

DR. JESÚS ROONEY RIVERA GUILLÉN (8943)
Vocal

DR. JUAN PRIMO BENÍTEZ RANGEL (2088)
Suplente

San Juan del Río, Querétaro, Agosto 2016.

Resumen

La calidad de la energía eléctrica es un factor de gran importancia para los procesos que hoy día se llevan a cabo en diferentes sectores de trabajo, en la industria se ve reflejado en parámetros como la calidad y tiempo de elaboración o servicios que esta ofrece. La calidad se ve afectada por distintas anomalías que de forma general son llamados disturbios eléctricos. Entre estos existen el sag y el swell que son variaciones de corta duración y son las que se estudian en este trabajo, en el que se desarrolla un sistema de detección para dichos disturbios. El sistema de detección está basado en una variante de una técnica moderna de optimización, llamada Micro Algoritmos Genéticos que está fundamentada en la teoría de Darwin sobre la evolución de las especies o selección natural, esta metodología se implementa en software usando MATLAB. Se realizaron pruebas en el sistema con señales sintéticas y reales adquiridas con los dos tipos de disturbios con la finalidad de validar la metodología descrita en las arquitecturas digitales. De manera conjunta se indujeron los dos tipos de disturbios eléctricos en una máquina fresadora CNC con la finalidad de analizar la afectación en los parámetros de movimiento de la máquina ante estos disturbios.

(Palabras clave: Calidad de la Energía, Sag, Swell Micro Algoritmos Genéticos)

Agradecimientos

A mi familia por el apoyo y respaldo incondicional que me han brindado a lo largo de mi formación personal y académica.

A los profesores, compañeros y amigos que he tenido la oportunidad de conocer hasta este momento de mi vida, de alguna manera cada uno me enseñó o aprendí algo de ellos y por los buenos momentos y recuerdos que compartimos.

Índice General

Resumen	1
Agradecimientos	2
Índice General.....	3
Índice de figuras	5
Índice de tablas	8
Capítulo 1	9
Introducción.....	9
1.1 Antecedentes	9
1.2 Objetivos	11
1.2.1 Objetivo General.....	11
1.2.2 Objetivos Particulares.....	11
1.3 Descripción del Problema	12
1.4 Justificación	12
1.5 Planteamiento general del proyecto	13
Capítulo 2	15
Revisión de la literatura.....	15
2.1 Dispositivos FPGA	15
2.2 Calidad de la energía eléctrica	17
2.3 Disturbios eléctricos	17
2.3.1 Variaciones de corta duración	18
2.3.1.1 Interrupción.....	19
2.3.1.2 Sag	19
2.3.1.3 Swell	19
2.4 Máquinas-herramienta	21

2.4.1	Máquinas CNC	21
2.5	Dinámica de movimiento.....	24
2.6	Algoritmos genéticos	27
2.6.1	Representación de las variables de diseño.....	28
2.6.2	Operadores genéticos.....	28
2.6.3	Micro-Algoritmos Genéticos, MAG.....	30
Capítulo 3	31
Metodología	31
3.1	Descripción de la metodología basada en MAG	32
3.2	Implementación de la metodología en Software.....	33
3.3	Diseño de las arquitecturas para la implementación en Hardware	40
3.4	Plataforma de pruebas.....	45
Capítulo 4	50
Resultados	50
4.1	Resultados de la implementación en software	50
4.2	Resultados de la implementación en hardware.....	63
4.3	Resultados de pruebas sobre afectación de disturbios en máquina fresadora (Dinámica de Movimiento).....	71
Conclusiones y prospectivas	84
Referencias	85
Anexos	87

Índice de figuras

Figura 1.1. Planteamiento general del proyecto.	14
Figura 2.1. Estructura conceptual de un dispositivo FPGA.	15
Figura 2.2. Diagrama conceptual de una LUT de 3 entradas basada en la celda lógica.	16
Figura 2.3. Variación de corta duración Sag.	20
Figura 2.4. Variación de corta duración Swell.	20
Figura 2.5. Torno CNC (Fuente: www.tecnomaquinaria.com).	22
Figura 2.6. Máquina fresadora CNC (Fuente: www.tecnomaquinaria.com).	23
Figura 2.7. Centro de maquinado CNC (Fuente: www.tecnomaquinaria.com).	23
Figura 2.8. Perfil de posición en la naturaleza.	24
Figura 2.9. Perfil de velocidad en la naturaleza.	25
Figura 2.10. Perfil de aceleración en la naturaleza.	26
Figura 2.11. Perfil de jerk en la naturaleza.	26
Figura 2.12. Proceso de un Algoritmo Genético	29
Figura 3.1. Diagrama general de la metodología seguida.	31
Figura 3.2. Diseño de las arquitecturas digitales para la implementación en Hardware	40
Figura 3.3. Bloque Población Inicial de las arquitecturas en VHDL.	41
Figura 3.4. Bloque Cordic de las arquitecturas en VHDL.	41
Figura 3.5. Bloque LUT_Seno de las arquitecturas en VHDL.	42
Figura 3.6. Bloque Generador de Error de las arquitecturas en VHDL.	42
Figura 3.7. Bloque Ordenamiento de las arquitecturas en VHDL.	43
Figura 3.8. Bloque Operadores Genéticos de las arquitecturas en VHDL.	44
Figura 3.9. Bloque Contador de Iteraciones de las arquitecturas en VHDL.	44
Figura 3.10. Bloque Indicador de Disturbios de las arquitecturas en VHDL.	45
Figura 3.11. Bloque Jerárquico de las arquitecturas en VHDL.	45
Figura 3.12. Diagrama de conexión del banco de pruebas (generación de disturbios)	46
Figura 3.13. Sistema propietario de generación de disturbios eléctricos sag y swell junto con el sistema de adquisición de datos.	47
Figura 3.14. Sistema de adquisición de datos.	48
Figura 3.15. Máquina fresadora donde se realizaron las pruebas.	49

Figura 4.1. Señal Original con disturbio para señales sintéticas: (a) Sag y (b) Swell.....	51
Figura 4.2. Señal Original con disturbio para señales de base de datos: (a) Sag y (b) Swell.	52
Figura 4.3. Señal fundamental para señales sintéticas: (a) Sag y (b) Swell.	54
Figura 4.4. Señal fundamental para señales de base de datos: (a) Sag y (b) Swell.	55
Figura 4.5. Detector de fase para señales sintéticas: (a) Sag y (b) Swell.	56
Figura 4.6. Detector de fase para señales de base de datos: (a) Sag y (b) Swell.....	57
Figura 4.7. Análisis en la señal fundamental para señales sintéticas: (a) Sag y (b) Swell. .	59
Figura 4.8. Análisis en la señal fundamental para señales de base de datos: (a) Sag y (b) Swell.	60
Figura 4.9. Señal Envolvente en la señal fundamental para señales sintéticas: (a) Sag y (b) Swell.	61
Figura 4.10. Señal Envolvente en la señal fundamental para señales de base de datos: (a) Sag y (b) Swell.	62
Figura 4.11. Simulación de las arquitecturas digitales para un disturbio sag.....	63
Figura 4.12. Señal envolvente en la simulación de las arquitecturas digitales para un disturbio sag.....	64
Figura 4.13. Evolución del método de MAG en la simulación de las arquitecturas digitales.	65
Figura 4.14. Convergencia del método de MAG en la simulación de las arquitecturas digitales.....	65
Figura 4.15. Evolución del método de MAG en la simulación de las arquitecturas digitales en un disturbio sag.....	66
Figura 4.16. Convergencia del método de MAG en la simulación de las arquitecturas digitales en un disturbio sag	67
Figura 4.17. Simulación de las arquitecturas digitales para un disturbio swell.....	68
Figura 4.18. Señal envolvente en la simulación de las arquitecturas digitales para un disturbio swell	69
Figura 4.19. Convergencia del método de MAG en la simulación de las arquitecturas digitales en un disturbio swell	70

Figura 4.20. Evolución del método de MAG en la simulación de las arquitecturas digitales en un disturbio swell.....	70
Figura 4.21. Adquisición de las señales de voltaje.....	71
Figura 4.22. Adquisición de las señales de voltaje con un disturbio sag	72
Figura 4.23. Adquisición de las señales de voltaje con un disturbio swell	73
Figura 4.24. Señales de referencia de los parámetros de movimiento de la máquina fresadora	74
Figura 4.25. Señales de seguimiento de los parámetros de movimiento de la máquina fresadora	75
Figura 4.26. Señales de seguimiento de los parámetros de movimiento de la máquina fresadora en el momento de un disturbio Sag de 90% de amplitud	76
Figura 4.27. Señales de seguimiento de los parámetros de movimiento de la máquina fresadora en el momento de un disturbio Sag de 60% de amplitud	77
Figura 4.28. Señales de seguimiento de los parámetros de movimiento de la máquina fresadora en el momento de un disturbio Sag de 75% de amplitud	78
Figura 4.29. Señales de seguimiento de los parámetros de movimiento de la máquina fresadora en el momento de un disturbio Swell de 110% de amplitud	79

Índice de tablas

Tabla 2.1. Clasificación de las variaciones de corta duración.....	18
Tabla 4.1. Valores estadísticos para las pruebas realizadas en el parámetro de posición	80
Tabla 4.2. Valores estadísticos para las pruebas realizadas en el parámetro de velocidad ..	81
Tabla 4.3. Valores estadísticos para las pruebas realizadas en el parámetro de aceleración	82
Tabla 4.4. Valores estadísticos para las pruebas realizadas en el parámetro de jerk.....	83

Capítulo 1

Introducción

Las máquinas-herramientas CNC son utilizadas hoy día en una enorme cantidad de procesos industriales, en ello se ve involucrada la calidad del proceso y la calidad de la pieza manufacturada, por consiguiente esto indica que las máquinas-herramientas deben estar en óptimas condiciones, lo cual no siempre es posible debido a que existen anomalías que afectan la calidad de la energía eléctrica que pueden llegar a modificar los parámetros de movimiento de la máquina y por consiguiente afectar el resultado final en la pieza, esto se ve reflejado en grandes pérdidas monetarias en la industria incluso verse obligada a interrumpir sus procesos. En la actualidad existen diversos sistemas para monitorear y detectar cuando se presentan disturbios o fallas en la red eléctrica los cuales generalmente son medidores comerciales, pero en específico para las máquinas herramienta no siempre se cuenta con un sistema de detección de este tipo, en el desarrollo de este proyecto se describe un sistema de detección de algunos disturbios en una máquina fresadora CNC (*Computer Numeric Control*; control numérico computarizado) , este sistema es descrito en FPGA (*Field Programmable Gate Array*; arreglo de compuertas programables en campo) ya que este dispositivo permite gran facilidad en el manejo de procesamiento de señales, la metodología que se utilizará para el sistema de detección es una variante de un método de optimización que hoy día es utilizado para resolver problemas en ingeniería llamado Algoritmos Genéticos (AG), también se realiza un análisis de la dinámica de movimiento de la máquina al verse afectado por los disturbios eléctricos.

1.1 Antecedentes

- Aportaciones sobre dinámica de movimiento en maquinaria

Los trabajos que se han presentado sobre el tema se enfocan a mejorar el desempeño de las máquinas-herramienta. Trejo (2006) desarrolló un módulo basado en software para realizar monitoreo y maquinado en un proceso de torneado. Osornio (2007) realizó el diseño de un sistema de control, basado en FPGA para máquinas-herramientas CNC de alta velocidad. Morales et al (2010) presentaron una arquitectura abierta basada en

FPGA bajo una filosofía de control distribuida multi-agente aplicado a sistemas de control en máquinas CNC. Osornio et al (2007) presentaron una técnica para la generación de perfiles polinomiales de aceleración- desaceleración en tiempo discreto, basado en un análisis paramétrico computacionalmente eficiente en una aplicación basada en FPGA. Tiempo después de Santiago et al (2008) presentaron un método basado en FPGA que reconstruía la dinámica a partir de información proporcionada por un encoder incremental óptico, sin embargo, la reconstrucción se llevó a cabo fuera de línea. El control de la dinámica se logra mediante el diseño de perfiles polinomiales, existen diferentes técnicas para la generación de perfiles polinomiales como la parametrización de perfiles polinomiales a trozos (Rivera, 2010).

- Aportaciones sobre calidad de la energía y disturbios eléctricos

El monitoreo de la calidad de la energía es una herramienta que permite realizar análisis y diagnóstico, tanto técnicos como económicos en la industria, proporcionando soluciones a problemas generados por la presencia de anomalías eléctricas. Granados (2013) se enfoca en el monitoreo y procesamiento de la señales de voltaje y corriente de una máquina CNC para conocer las condiciones de calidad de la energía, García (2014) propone diferentes metodologías para sensores inteligentes que se enfocan principalmente a la detección de fallas en máquinas eléctricas bajo diferentes criterios de operación. En lo relacionado a los disturbios eléctricos, Granados et al (2013) proponen un sensor de detección, clasificación y cuantificación de los disturbios en la calidad de la energía, usando como técnica de detección la transformada Hilbert, Radil et al (2008) aplicaron diferentes algoritmos para detectar y clasificar los disturbios que se producían en un sistema eléctrico.

- Aportaciones sobre Algoritmos Genéticos

Las investigaciones relacionadas con los algoritmos genéticos se han vuelto novedosas porque presentan soluciones a problemas complejos de ingeniería y porque tienen aplicaciones en muchos campos de estudio, algunas se han hecho sobre la optimización de valores. Pérez (2009) desarrolló el diseño óptimo de una microviga utilizando esta técnica. En el área de control también han sido utilizados para escoger los

parámetros del controlador, Luna (2010) diseñó e implementó en FPGA un algoritmo genético para la sintonización de controladores PID. Jaen (2014) utilizó los algoritmos genéticos para optimizar la sintonización de los controladores en sistemas de posicionamiento. En cuanto a la detección de disturbios eléctricos los algoritmos genéticos se han utilizado solamente como un recurso de soporte para diferentes metodologías sin llegar a ser la base del sistema de detección.

La Universidad Autónoma de Querétaro cuenta con trabajos publicados en revistas internacionales por parte de profesores titulares, además de varios trabajos de tesis relacionados con los temas, por lo que se puede indicar que los trabajos realizados en esta institución tienen gran impacto en el ámbito internacional.

1.2 Objetivos

1.2.1 Objetivo General

Diseñar un sistema de detección de disturbios Sag y Swell en fresadora CNC implementando Micro Algoritmos Genéticos en arquitecturas FPGA para el análisis de su impacto en su dinámica de movimiento.

1.2.2 Objetivos Particulares

- Diseñar un método de detección de Sag y Swell para señales sintéticas usando Micro Algoritmos Genéticos mediante código en MATLAB.
- Inducir sobre una máquina fresadora disturbios Sag y Swell por medio de un sistema propietario enfocado a la generación de disturbios el cual, es integrado por elementos comerciales de potencia.
- Implementar la metodología propuesta para la detección de los disturbios fuera de línea para verificar su funcionalidad sobre señales reales.
- Integrar los módulos en hardware necesarios para la implementación de la metodología de detección en una tarjeta basada en FPGA para la detección de disturbios en línea usando la plataforma de pruebas desarrollada.

- Realizar pruebas de perfiles de movimiento sometidos a los dos disturbios para adquirir parámetros dinámicos de la fresadora y mediante procesamiento de señales analizar la afectación de dichos parámetros ante el disturbio.

1.3 Descripción del Problema

Los sistemas industriales cuentan con sistemas de detección de fallas o disturbios en la red eléctrica para toda la planta, los cuales es importante resaltar son medidores portátiles, sin embargo en las máquinas-herramientas no se tienen sistemas fijos de detección ante disturbios eléctricos poniendo así en riesgo el proceso de manufactura y con esto la calidad de la producción.

Los sistemas de medición con los que se cuentan en la actualidad son de arquitectura cerrada, y por lo tanto no permiten modificaciones adicionales al sistema, esto también los hace de gran costo.

Los sistemas de detección de fallas tienen en su procesamiento de señales una metodología muy variada. Entre estas metodologías no es muy común encontrar una basada en algoritmos genéticos para la detección de disturbios eléctricos.

En los trabajos realizados, al momento, no se ha analizado como se ve afectada la dinámica de movimiento de la máquina ante los disturbios eléctricos.

Por lo general, en las microempresas y pequeñas empresas no cuentan con este tipo de sistemas de detección y, por lo tanto, sus procesos de manufactura pueden verse mayormente afectados.

1.4 Justificación

Detectar de manera individual para cada máquina-herramienta CNC los disturbios que se presenten en la calidad de la energía.

El sistema de detección de disturbios eléctricos será de arquitectura abierta, dando apertura a posibles modificaciones en el futuro y de fácil acceso a ser reconfigurable.

Mediante la metodología de algoritmos genéticos se realizará un sistema de detección de disturbios eléctricos en una máquina fresadora.

Análisis de la afectación en el movimiento de la máquina ante los dos tipos de disturbios eléctricos. Este desarrollo no se ha analizado de una manera más profunda.

Contribuir con el desarrollo de un módulo para ofrecer soluciones tecnológicas a pequeñas y medianas empresas.

1.5 Planteamiento general del proyecto

El planteamiento general de este proyecto se desarrolla mediante el diagrama a bloques mostrado en la Figura 1.1. *Error! No se encuentra el origen de la referencia.* El diagrama se divide en dos bloques principales: uno de detección y otro de análisis; en el bloque de detección se encuentran dos aspectos: la metodología basada en Algoritmos Genéticos, en la cual a través de los distintos bloques se va creando el sistema de detección de disturbios sag y swell partiendo desde una validación en software, y sobre señales reales para después implementar el sistema de detección en una plataforma en FPGA. El otro aspecto es la calidad de la energía, en esta parte se cuenta con el generador de disturbios eléctricos sag y swell propietario el cual se utilizará para inducir los dos tipos de disturbios en la máquina fresadora en la cual se realizara el estudio. En el bloque de análisis se tiene contemplado el tema de dinámica de movimiento, en esta parte se obtienen las señales de la máquina fresadora en el instante del disturbio eléctrico mediante un codificador de posición (encoder), estos disturbios inducidos en la máquina afectarán los parámetros de movimiento de la máquina que mediante un procesamiento digital de las señales permitirán realizar un análisis de la afectación en la dinámica de movimiento ante los disturbios eléctricos.

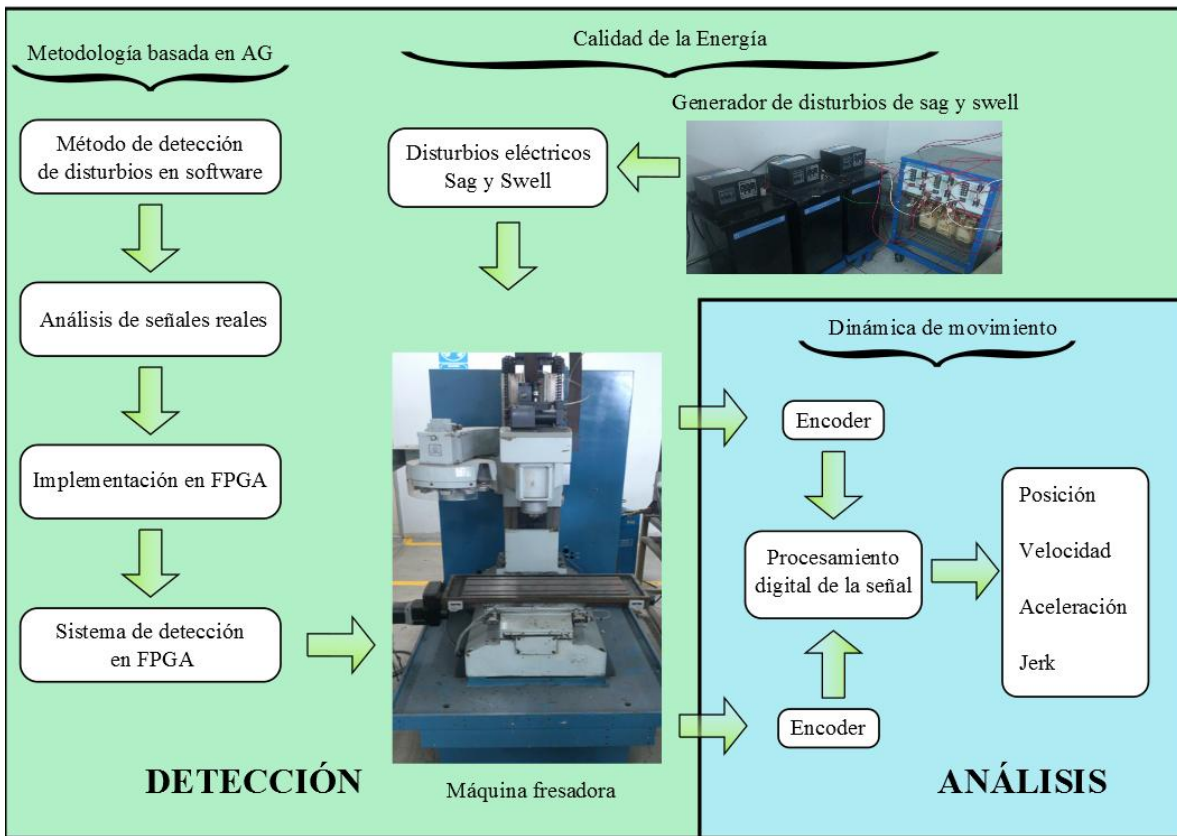


Figura 1.1. Planteamiento general del proyecto.

Capítulo 2

Revisión de la literatura

2.1 Dispositivos FPGA

Desde los comienzos del hardware de lógica digital en los años 70's, han existido una gran cantidad de dispositivos, conduciéndonos a la serie TTL o la tecnología CMOS, a pesar que estos dispositivos son aún usados frecuentemente en el diseño de circuitos impresos, ha existido un esfuerzo constante los últimos 20 años para introducir mayor programabilidad dentro de los dispositivos digitales básicos.

Como resultado, los dispositivos programables se desarrollaron como una forma de aproximación intermedia; diseño en hardware de alto rendimiento y recursos óptimos, no se requiere de ningún sistema operativo y el dispositivo puede ser reprogramable.

Un FPGA es un dispositivo lógico que contiene un arreglo bi-dimensional de celdas lógicas genéricas e interruptores programables. La estructura conceptual de un dispositivo FPGA se muestra en la Figura 2.1.

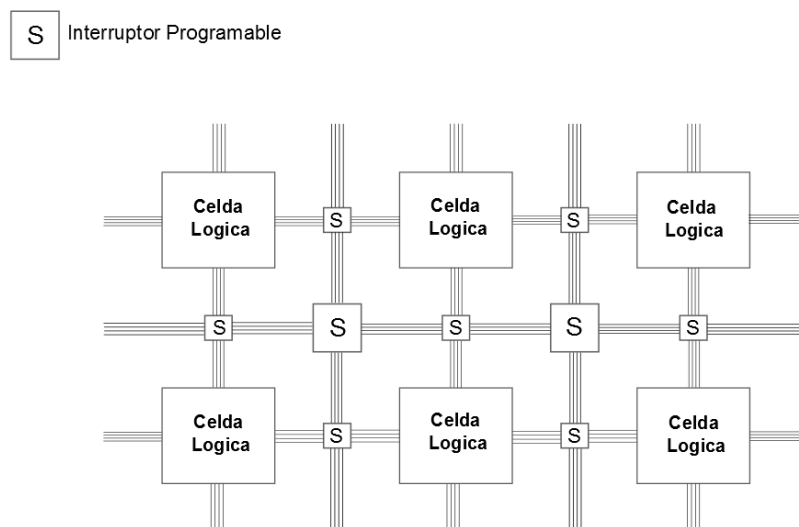


Figura 2.1. Estructura conceptual de un dispositivo FPGA.

Una celda lógica usualmente contiene un pequeño circuito combinacional configurable con un Flip-Flop tipo D. El método más común para implementar un circuito combinacional configurable es una LUT (*Look-up Table*; tabla de búsqueda). Una LUT puede ser configurada para dar un tipo específico de función lógica cuando se programa. En la Figura 2.2 se muestra el diagrama conceptual de una LUT de 3 entradas basada en la celda lógica. Se puede notar que la salida de la LUT puede ser usada directamente o almacenada en el Flip-Flop tipo D. Este último puede ser usado para implementar circuitos secuenciales.

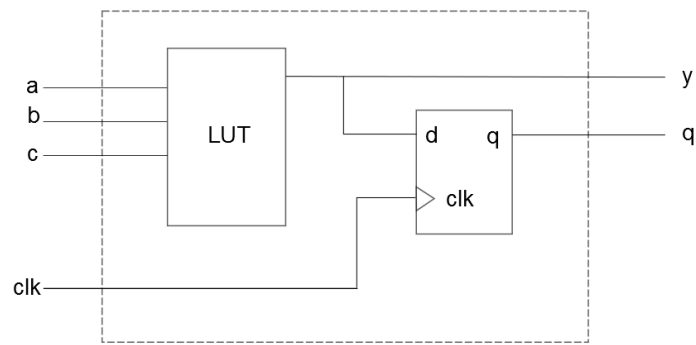


Figura 2.2. Diagrama conceptual de una LUT de 3 entradas basada en la celda lógica.

Una celda lógica puede ser configurada para llevar a cabo una función simple, y un interruptor programable puede ser personalizado para proporcionar interconexiones entre las celdas lógicas. Un diseño personalizado puede implementarse especificando la función de cada celda lógica y selectivamente estableciendo la conexión de cada interruptor programable. Una vez que el diseño y la síntesis se han completado, se puede utilizar un cable adaptador para descargar la configuración en las celdas lógicas y los interruptores deseados en el dispositivo FPGA y obtener el circuito personalizado. Dado que este proceso puede realizarse “en el campo” (*“in the field”*) en lugar de ser fabricado, el dispositivo es conocido como *Field Programmable*.

Los FPGAs son dispositivos verdaderamente revolucionarios que combinan los beneficios de hardware y software. Estos implementan circuitos al igual que el hardware,

proporcionando enorme capacidad, área y beneficios de rendimiento sobre el software, también pueden ser reprogramados de forma barata y sencilla para implementar una amplia gama de tareas. Algunas veces la reprogramación es solo para corregir un error en algún mal comportamiento o para añadir una nueva característica. Normalmente, los FPGAs son útiles para las operaciones que procesan grandes flujos de datos, como procesamiento de señales, redes, etc., (Hauck y Dehon, 2008).

Cabe mencionar que este tipo de dispositivo es el que ha ganado un mayor mercado en las aplicaciones modernas de diseño digital y se pueden contar con unidades que van desde los 50,000 hasta los 10, 000,000 de compuertas lógicas equivalentes (Romero, 2007).

2.2 Calidad de la energía eléctrica

La calidad de la energía eléctrica se entiende cuando la energía eléctrica es suministrada a los equipos y dispositivos con las características y condiciones adecuadas que les permita mantener su continuidad sin que se afecte su desempeño ni provoque fallas a sus componentes.

Se puede decir que existe un problema de calidad de la energía eléctrica cuando ocurre cualquier desviación de la tensión, la corriente o la frecuencia que provoque la mala operación de los equipos y deteriore la economía o el bienestar de los usuarios. Cualquier desviación de este tipo es un disturbio de la calidad de la energía.

El incremento en la aplicación de equipos electrónicos puede causar perturbaciones electromagnéticas o que pueden ser sensibles a estos fenómenos, esto ha aumentado el interés por la calidad de la energía en los últimos años.

2.3 Disturbios eléctricos

La Norma IEEE Std. 1159-2009 provee los pasos para el monitoreo del sistema eléctrico monofásico y polifásico de corriente alterna y la guía para un apropiado entendimiento de los resultados obtenidos, además describe los diversos tipos de fenómenos electromagnéticos que se presentan en las redes eléctricas. Los diferentes tipos de perturbaciones o fenómenos electromagnéticos que pueden ocurrir en un sistema

eléctrico de potencia y que se consideran en el estudio de la calidad de la energía están basados en la Norma IEEE Std. 1159-2009 (IEEE Recommended Practice for Monitoring Electric Power Quality). Básicamente, la clasificación de los fenómenos electromagnéticos se da en siete categorías, según sean las características de su espectro armónico, su duración y su amplitud. Se tienen entonces fenómenos transitorios, variaciones de corta duración, variaciones de larga duración, desbalances de tensión, distorsión de la forma de onda, fluctuaciones de tensión y variaciones de la frecuencia, (Granados, 2013).

A continuación, se describen los disturbios que son variaciones de corta duración que son los que nos interesan en este trabajo.

2.3.1 Variaciones de corta duración

Son eventos de la red eléctrica con variaciones en la tensión. Dependiendo de la localización de la falla y las condiciones del sistema, la falla puede causar aumentos en la tensión (swell), decrementos en la tensión (sag) o una completa pérdida de la tensión (interrupción); en cualquier caso, el impacto en la tensión durante la condición de falla es una variación de corta duración. Cada tipo de variación puede ser designada como instantánea, momentánea o temporal dependiendo de su duración. En la Tabla 2.1 se muestra la clasificación de estos eventos dependiendo su duración de acuerdo al estándar IEEE std. 1159-2009.

Tabla 2.1. Clasificación de las variaciones de corta duración

Categoría		Duración	Magnitud de voltaje (p.u.)
Instantánea	Sag	0.5 a 30 ciclos	0.1 a 0.9
	Swell	0.5 a 30 ciclos	1.1 a 1.8
Momentánea	Interrupción	0.5 a 3 segundos	< 0.1
	Sag	30 ciclos a 3 segundos	0.1 a 0.9
	Swell	30 ciclos a 3 segundos	1.1 a 1.4
Temporal	Interrupción	3 segundos a 1 minuto	< 0.1
	Sag	3 segundos a 1 minuto	0.1 a 0.9
	Swell	3 segundos a 1 minuto	1.1 a 1.2

2.3.1.1 Interrupción

Una interrupción ocurre cuando el voltaje de alimentación o la corriente disminuyen a un valor menor de 1 p.u. (por unidad) por un periodo de tiempo que no excede un minuto. Pueden ser resultado de fallas en el sistema de energía, fallas en los equipos o mal funcionamiento de un control. Las interrupciones son medidas por su duración desde que la magnitud de la tensión es menor del 10% de la nominal.

2.3.1.2 Sag

Un sag es un decremento en la tensión nominal entre 0.1 p.u. y 0.9 p.u. para duraciones de 0.5 ciclos hasta 1 minuto. Estos son usualmente asociados con condiciones de falla del sistema pero también pueden ser causados por energizar grandes cargas o arrancar grandes motores. En la Figura 2.3 se muestra un disturbio sag.

2.3.1.3 Swell

Un swell es un incremento en la tensión nominal por encima de 1.1 p.u. para duraciones de 0.5 ciclos hasta 1 minuto. Como los sags, los swells son asociados con condiciones de falla del sistema pero son menos comunes que los sags. Una falla de línea a tierra puede ocasionar un swell, también pueden ser causados por la desenergización de grandes cargas o la energización de un gran banco de capacitores. En la Figura 2.4 se muestra un disturbio swell.

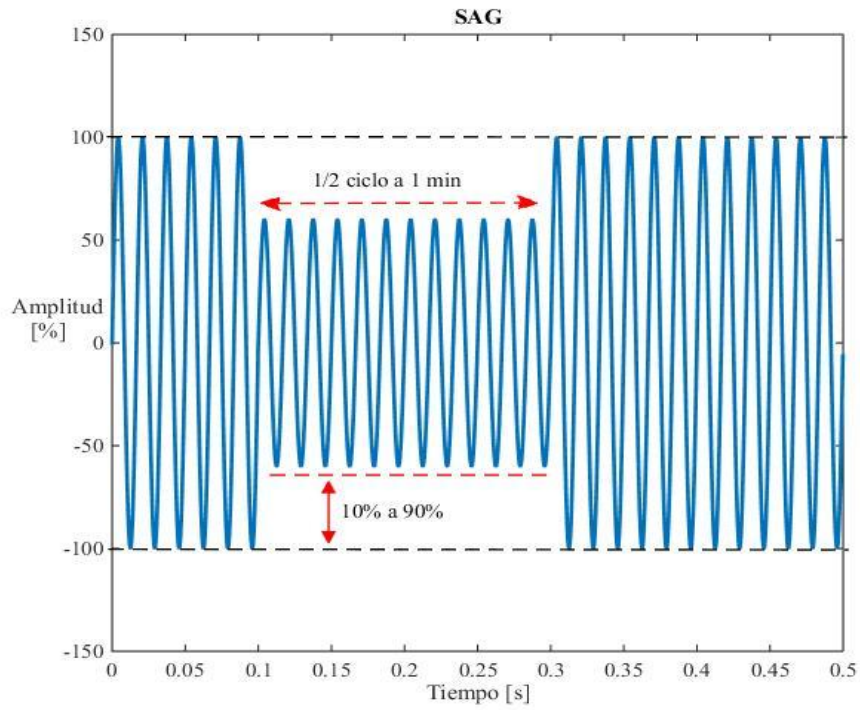


Figura 2.3. Variación de corta duración Sag.

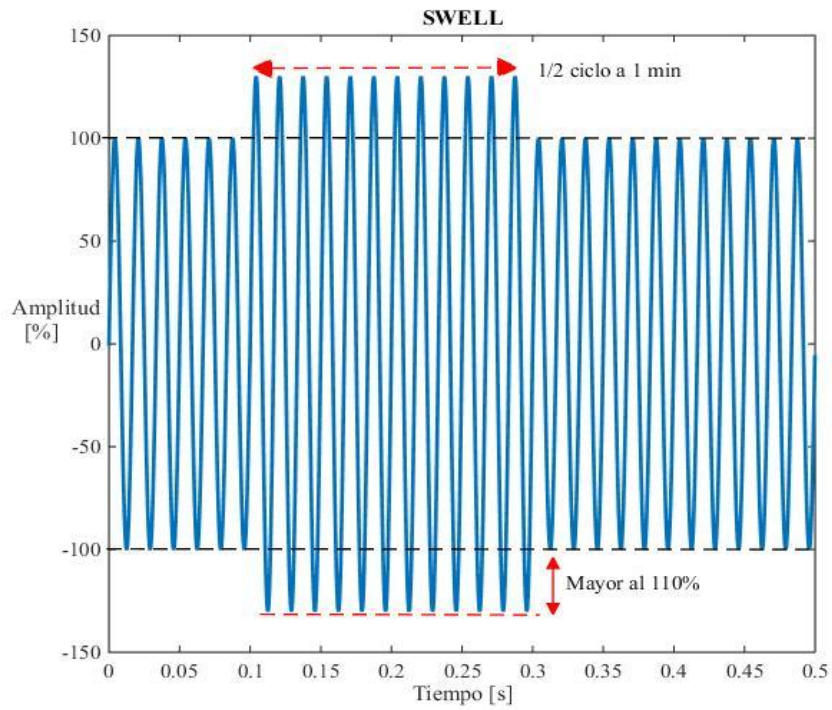


Figura 2.4. Variación de corta duración Swell.

2.4 Máquinas-herramienta

Las máquinas-herramienta son dispositivos para el corte de materiales (principalmente metales), para darles la forma requerida. Estas cuentan con un arreglo que facilita el uso de varios tipos de herramientas de corte que pueden ser cambiadas para adecuarse a la tarea requerida, así mismo, pueden ser afiladas nuevamente. Las máquinas-herramienta pueden ser clasificadas de muchas maneras; de acuerdo a su tamaño: de trabajo ligero, trabajo medio y trabajo pesado; de acuerdo al método de actuación: operadas manualmente, semiautomáticas y completamente automatizadas; de acuerdo a su propósito: las que son usadas para una amplia variedad de operaciones y las que son para operaciones específicas que tienen un limitado rango de formas y piezas de trabajo; otra clasificación es de acuerdo con los tipos de movimientos para remover el material. Una buena máquina-herramienta debe tener alta productividad y adecuada precisión con las especificaciones de la pieza de trabajo y un buen control de las palancas, botones, etc., en una posición conveniente para el operador, debe garantizar la seguridad del operador y de la máquina y sus gastos de fabricación y funcionamiento deben ser razonables.

2.4.1 Máquinas CNC

Una máquina-herramienta es llamada “máquina madre” en el sentido de que es una máquina que crea máquinas. En particular, las máquinas-herramienta han avanzado de ser máquinas-herramienta manuales a ser máquinas de control numérico, estas se han perfeccionado en el rol de ser máquinas madre con gran precisión y velocidad de maquinado.

Las máquinas NC (*Numeric Control*; control numérico) fueron desarrolladas con el propósito de maquinar partes con formas complejas y de una manera precisa. El control numérico fue primeramente aplicado a máquinas fresadoras, pero recientemente para mejorar la productividad se extendió a diversos tipos de máquinas-herramienta. Los sistemas de CN no solo son usados por máquinas-herramienta, también lo utilizan máquinas que necesitan movimiento controlado por servosistemas.

El CN es el sistema que habilita la máquina-herramienta para maquinar partes con distintas formas de manera rápida y precisa. En CN el servomotor es usado para controlar la

máquina-herramienta de acuerdo a la operación del usuario. Esto es, el CN es un dispositivo de control que maquina una pieza objetivo activando el servomotor de acuerdo con los comandos establecidos. El CN combinado con tecnología computacional es llamado control numérico por computadora o CNC. En lugar de elementos y circuitos, las funciones de CN son implementadas basadas en software en un CNC.

Actualmente diversas máquinas-herramienta cuentan con un sistema CNC, entre la principal maquinaria CNC tenemos las siguientes:

Torno: se utiliza para producir piezas redondas. La pieza de trabajo, sostenida por un dispositivo de sujeción montado en el eje del torno, se hace girar contra la herramienta de corte, que se mueve paralelamente al eje central de la pieza de trabajo, lo que produce una forma cilíndrica. Esta operación también es usada para producir superficies cilíndricas externas. En la Figura 2.5 se muestra la imagen de un torno CNC.



Figura 2.5. Torno CNC (Fuente: www.tecnomaquinaria.com).

Fresadora: se utiliza mayormente para generar superficies planas o complejas usando cortadores con múltiples filos cortantes denominados fresas. La pieza de trabajo que debe quedar sujeta en una prensa de tornillo, o con un aditamento o accesorio para fijar a la mesa, es avanzada o alimentada hacia la herramienta de corte giratoria. La fresadora está clasificada como horizontal o vertical. En la Figura 2.6 se muestra la imagen de una máquina fresadora CNC.



Figura 2.6. Máquina fresadora CNC (Fuente: www.tecnomaquinaria.com).

Centro de maquinado: puede funcionar ya sea como un torno o una fresadora, la función depende de los requerimientos del usuario y de la herramienta que se seleccione, ya que combina diversos tipos de herramienta de corte, el cambio lo realiza de forma automática el CNC. En la Figura 2.7 se muestra la imagen de un centro de maquinado CNC.



Figura 2.7. Centro de maquinado CNC (Fuente: www.tecnomaquinaria.com).

2.5 Dinámica de movimiento

El movimiento de todo sistema mecánico está sujeto a efectos físicos que influyen directamente en su comportamiento dinámico, estos efectos pueden ser analizados a través del comportamiento en posición, velocidad, aceleración y jerk que en conjunto se definen como la dinámica de movimiento del sistema mecánico. La forma en que se realiza el movimiento se le conoce como perfil de movimiento. Un perfil de movimiento describe el comportamiento dinámico que tiene un sistema mecánico al moverse, (Rivera, 2007).

En la naturaleza existen perfiles de movimiento que cumplen con la característica de suavidad en el movimiento, esto es debido a que el comportamiento se encuentra bien distribuido a lo largo de todo el perfil, las siguientes figuras muestran los perfiles en la naturaleza y su comportamiento en la naturaleza.

La Figura 2.8 muestra el perfil de posición en la naturaleza, se observa de la figura que el inicio y el fin del movimiento son suaves.

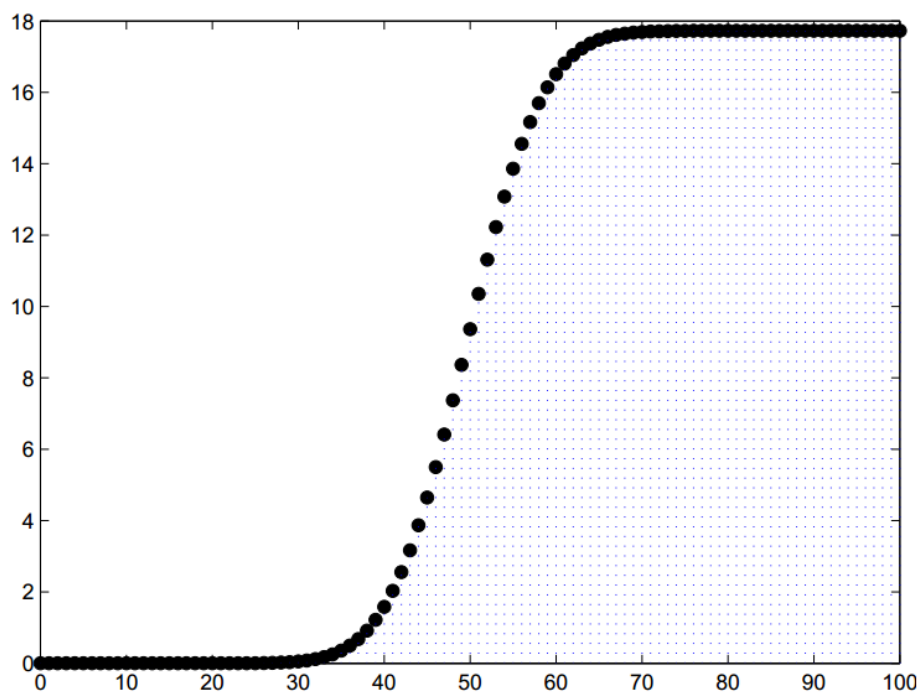


Figura 2.8. Perfil de posición en la naturaleza.

El perfil de velocidad se muestra en la Figura 2.9, se observa cómo se incrementa de forma gradual hasta llegar a su máximo y se va decrementando gradualmente hasta cero nuevamente, el pico de la gráfica y al final el movimiento es suave.

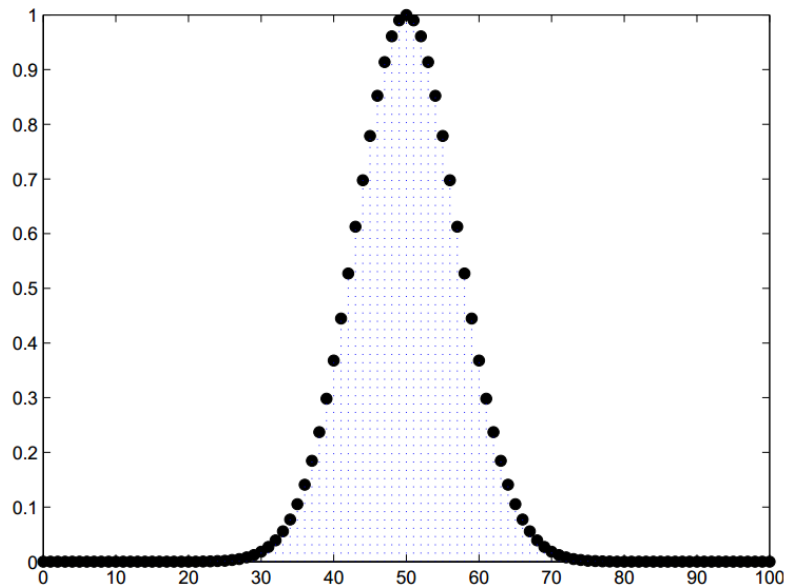


Figura 2.9. Perfil de velocidad en la naturaleza.

En el perfil de aceleración se observa que no se tienen cambios bruscos, Figura 2.10, los valores de aceleración inician y terminan en cero en una forma muy suave y el perfil se encuentra distribuido durante todo el movimiento.

En la Figura 2.11 se observa el perfil del jerk en la naturaleza, se tienen valores pequeños, completamente distribuidos durante todo el perfil, iniciando y terminando en cero, además de estar distribuido se observa que el movimiento es suave para todo el perfil.

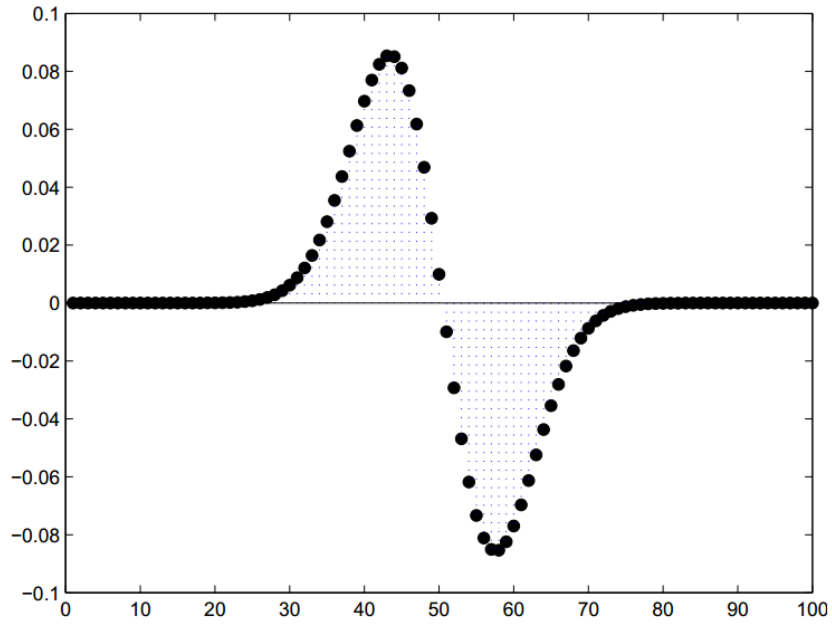


Figura 2.10. Perfil de aceleración en la naturaleza.

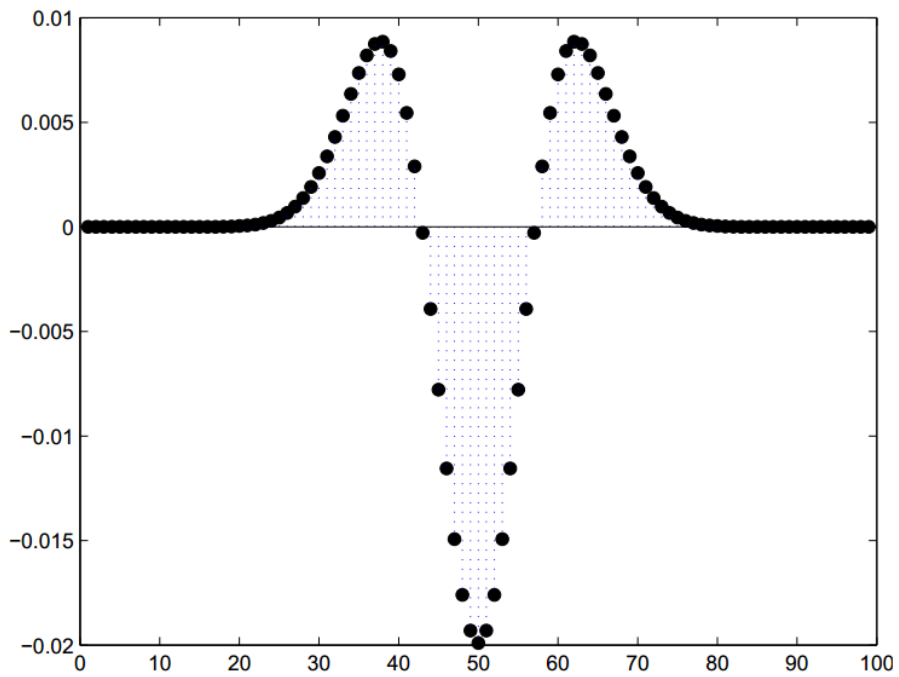


Figura 2.11. Perfil de jerk en la naturaleza.

2.6 Algoritmos genéticos

En años recientes, han sido desarrollados algunos métodos de optimización que son conceptualmente diferentes de las técnicas tradicionales de programación matemática. Estos métodos son etiquetados como métodos modernos de optimización. Muchos de estos métodos están basados en ciertas características y comportamientos como el biológico, molecular, enjambre de insectos y sistemas neurobiológicos. La mayoría de estos métodos se han desarrollado en los últimos años y están emergiendo como métodos populares para la solución de problemas complejos en ingeniería.

Los Algoritmos Genéticos (AG) están basados en la teoría de Darwin sobre la supervivencia del más apto, y tienen muchas ventajas sobre las técnicas clásicas de optimización. Por ejemplo, los AG no requieren de información basada en operaciones de gradiente que implican el uso de cálculo diferencial, cálculo integral, o el asumir condiciones iniciales específicas según el problema de optimización. Lo que se menciona en líneas anteriores significa que el uso de los AG puede ser extendido a espacios de búsqueda no solamente lineales, sino también no-lineales, así como no-convexos. Además de que por su naturaleza y estructura los AG pueden ser usados para la búsqueda de valores óptimos en problemas de una o múltiples variables, (Jaen, 2014).

Los algoritmos genéticos están basados en los principios de la genética y la selección natural, transponen las nociones de la evolución natural a las computadoras e imitan la evolución natural. Los algoritmos genéticos mantienen un conjunto de soluciones –población de individuos- y llevan a cabo una búsqueda multiobjetivo. Los individuos son representados por cromosomas compuestos de genes. Los algoritmos genéticos operan en los cromosomas, los cuales representan las propiedades heredables del individuo. Por analogía con la naturaleza, a través de la selección de los individuos más aptos (soluciones potenciales al problema de optimización), viven para reproducirse, y los individuos más débiles desaparecen. Los individuos se seleccionan y a partir de estos se crean nuevos individuos con operaciones de cruce y mutación, estos nuevos individuos reemplazan a los anteriores creando una nueva población, este proceso se repite hasta que toda la población se asemeja, momento en el cual se ha evolucionado hasta un valor nominal u óptimo.

2.6.1 Representación de las variables de diseño

En los algoritmos genéticos las variables de diseño son representadas como cadenas que son llamadas cromosomas. A cada cromosoma le corresponde un valor de la función objetivo conocido como adaptabilidad. La representación común es la cadena binaria de longitud fija. La elección de la longitud del cromosoma así como su codificación es la representación de las variables. Identificar una apropiada representación es el primer paso al resolver problemas de optimización usando algoritmos genéticos.

2.6.2 Operadores genéticos

La solución de un problema de optimización con AG empieza con una población de cadenas aleatorias que representan las variables de diseño. El tamaño de la población es usualmente fijo. Cada cadena (o variables de diseño) es evaluada para encontrar su valor de adaptabilidad en la función objetivo. La población es modificada por tres operadores: reproducción, cruza y mutación; con el objetivo de producir una nueva población. La nueva población es evaluada para encontrar sus valores de adaptabilidad y ver si converge en el proceso. Un ciclo de reproducción, cruza, mutación y evaluación de los valores de adaptabilidad es conocida como generación. Si el criterio de convergencia no se satisface, la población es iterativamente modificada por estos tres operadores y la nueva población generada es evaluada en sus valores de adaptabilidad. El procedimiento continúa a través de varias generaciones hasta que el criterio de convergencia es satisfecho y el proceso termina. A continuación se describen cada uno de estos operadores genéticos:

- **Reproducción:** es el primer operador aplicado a la población para seleccionar las mejores cadenas. Este operador también es llamado de selección. Una parte de la nueva población puede ser creada con solo copiar algunos individuos sin modificarlos. Esto nos da la posibilidad de supervivencia para las soluciones ya evaluadas.
- **Cruza:** el propósito de este operador es crear nuevos individuos en base intercambiar información entre dos cadenas. En este operador dos individuos son seleccionados y se intercambian algunas porciones de sus cadenas entre estos dos individuos. En el proceso se selecciona un punto de cruza y a partir de ahí se

intercambian las cadenas. Las dos cadenas seleccionadas para la cruce son llamadas padres y las cadenas generadas son llamadas descendencia. El punto de cruce puede ser escogido aleatoriamente. La descendencia generada puede o no puede ser tan buena como los padres en términos de su valor de adaptabilidad.

- **Mutación:** este operador es aplicado a las nuevas cadenas con cierta probabilidad de que ocurra. Este operador cambia uno o más valores en la representación, cuando es en un solo punto, un sitio a lo largo de la cadena es seleccionado aleatoriamente y se cambia su valor. El propósito de la mutación es: generar un individuo en las cercanías de la población, proteger contra una prematura pérdida de información del material genético en una determinada posición y mantener la diversidad en la población.

La Figura 2.12 muestra el proceso de un Algoritmo Genético

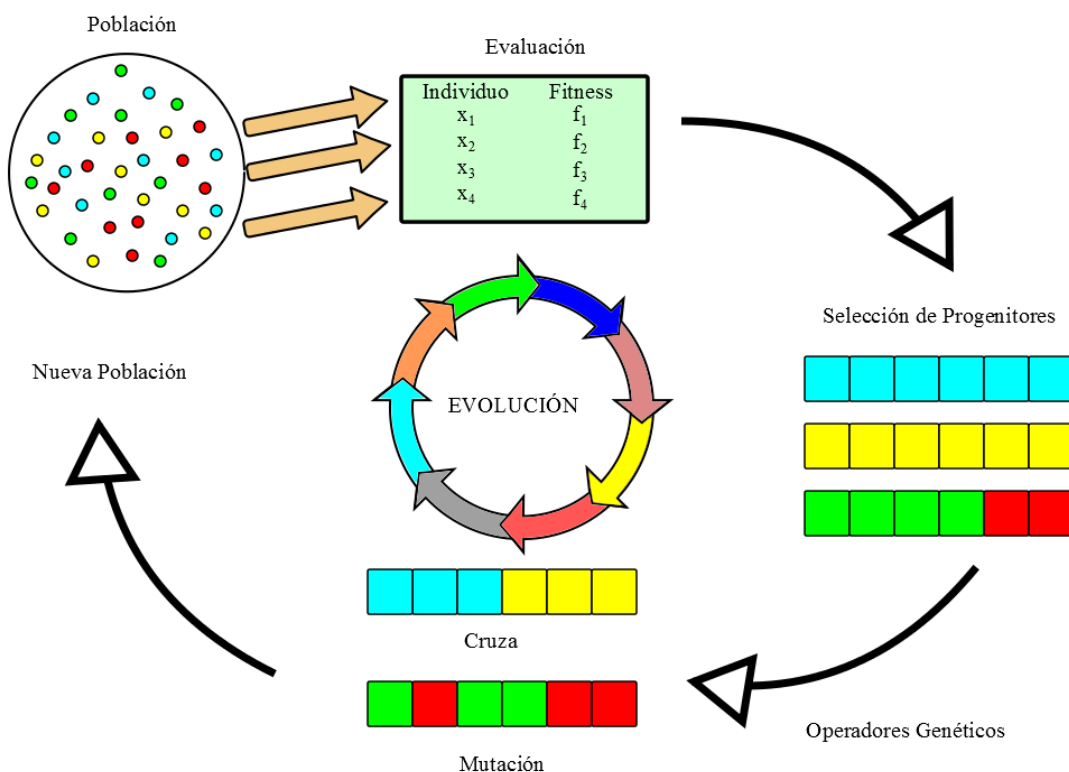


Figura 2.12. Proceso de un Algoritmo Genético

2.6.3 Micro-Algoritmos Genéticos, MAG

Hasta ahora ya se han mencionado las ventajas y características de los AG; por lo tanto, una forma de superar las limitaciones acerca de la gran cantidad de recursos que se demandan es aplicar el concepto de “micro-población”. La micro-población es una característica variante de los AG en la cual se define una cantidad muy pequeña de individuos que conforman la población del algoritmo.

De acuerdo con la literatura, una micro-población se puede componer desde cuatro hasta ocho o diez individuos, no se tiene un estándar específico como tal. Adicionalmente a la micro-población se pueden definir operadores genéticos de tal manera que una rápida convergencia y una búsqueda de los valores óptimos globales estén asegurados. Los algoritmos genéticos con una pequeña población disminuirán significativamente los recursos computacionales requeridos, ya que la cantidad de operaciones se minimizan al ser pocos individuos.

Capítulo 3

Metodología

En este capítulo se presenta el desarrollo de la metodología utilizada para la detección de disturbios eléctricos sag y swell, basada en una variante de la técnica meta-heurística de los AG, los cuales fueron descritos en el capítulo anterior, esta variante es conocida como Micro-Algoritmos Genéticos (MAG), y se describirá más a detalle en las siguientes secciones. La Figura 3.1 muestra el diagrama general de la metodología seguida en este trabajo para la realización y alcance de los objetivos planteados en el capítulo 1. A continuación se describen cada una de las etapas que conforman la metodología propuesta.

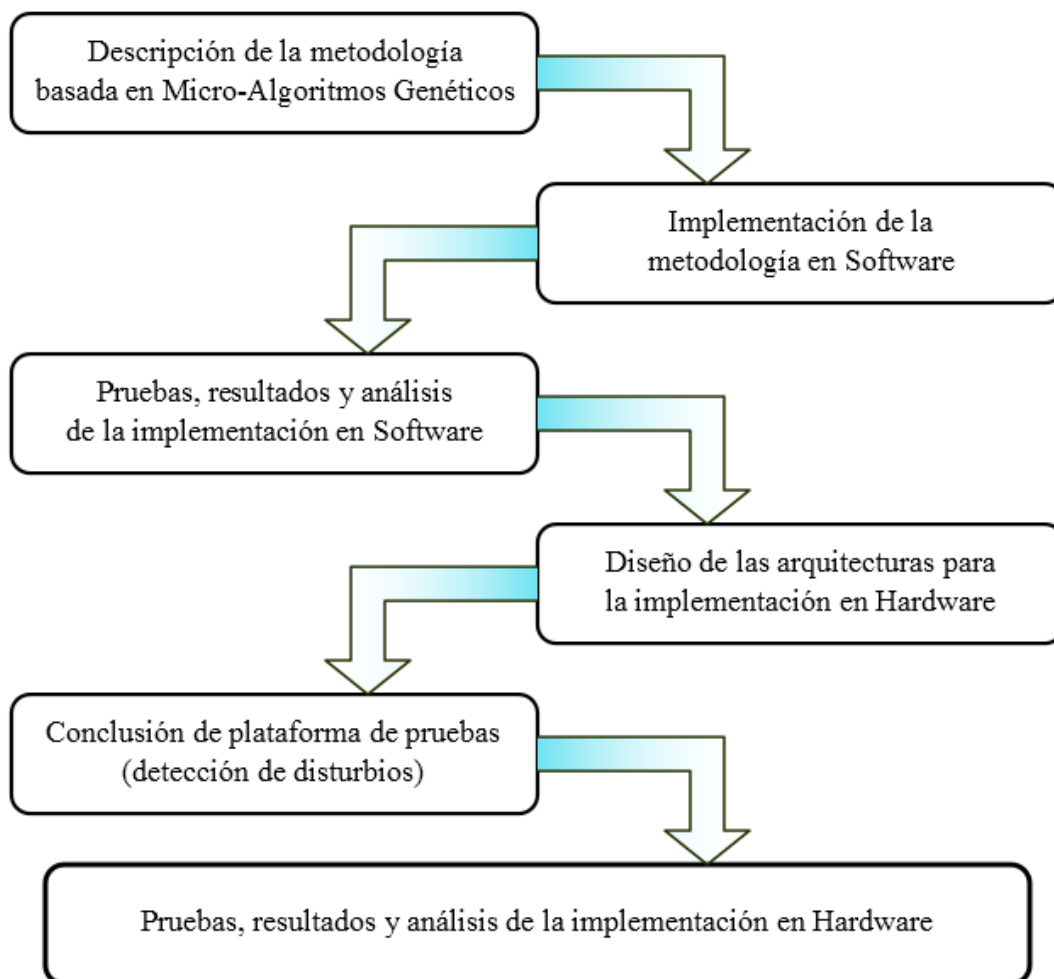


Figura 3.1. Diagrama general de la metodología seguida.

El diagrama consiste en varias etapas, en la primera se desarrolla la metodología basada en los MAG, una técnica meta-heurística usada para la búsqueda de valores óptimos en problemas de optimización de variables. En las etapas dos y tres se contempla la implementación de la metodología en software, así como las pruebas y los resultados que serán posteriormente analizados. La cuarta etapa consiste en el diseño de las arquitecturas digitales para la implementación en hardware de la metodología propuesta para la detección de disturbios eléctricos sag y swell. En la quinta etapa se validará el funcionamiento de la metodología propuesta, en esta etapa se realizará el banco de pruebas para la generación de los disturbios eléctricos para posteriormente inducirlos sobre la máquina fresadora CNC. En la última etapa se lleva a cabo la implementación en hardware del sistema basado en FPGA para la detección de disturbios eléctricos sag y swell usando MAG, así como el análisis de los resultados que produzca dicha implementación.

3.1 Descripción de la metodología basada en MAG

Los AG son una técnica de búsqueda de valores óptimos para un problema de optimización, y en donde un criterio de satisfacción condiciona la terminación del proceso iterativo de búsqueda de la solución al problema. La búsqueda de estos valores óptimos empieza definiendo aleatoriamente una cantidad específica de valores iniciales, los cuales representan posibles soluciones dentro del espacio de búsqueda. Los operadores genéticos adicionan variabilidad a la metodología de tal forma que se logra modificar a estas soluciones iniciales hasta que se convierten en el valor óptimo buscado.

En la metodología que se propone se utilizaron los Micro Algoritmos Genéticos en base a que respecto a los AG se diferencian en que se maneja el concepto de micro-población, donde las posibles soluciones son una cantidad mínima de valores iniciales, esto a su vez minimiza los recursos computacionales requeridos. En la presente investigación se define una población de solamente cuatro individuos.

Los rangos de diseño se especifican según el problema y las variables que se desean optimizar. La ecuación (1) representa una forma de onda pura de la línea eléctrica.

$$V(t) = V_m \sin(\omega t + \varphi) \quad (1)$$

Dónde:

$V(t)$ = Voltaje sinusoidal

V_m = Amplitud de la senoide

ω = Frecuencia angular en radianes

ωt = Argumento de la senoide

φ = Fase

En este escrito se define como variable de diseño el parámetro V_m de la ecuación (1), el parámetro V_m es la amplitud de la señal, el cual es de importancia porque los valores de este parámetro proporcionan la información necesaria para determinar si ocurrió un disturbio eléctrico ya sea un sag o un swell.

Después se realiza una correcta selección de los operadores genéticos que son tres principalmente: selección, cruce y mutación. Se define un proceso de selección elitista, la cruce se define como el promedio de la suma de los valores utilizados como padres y la mutación tiene un índice de probabilidad bajo.

Cada individuo es evaluado y por lo tanto tiene un desempeño, este desempeño se cuantifica al asignarle al individuo un valor de adaptabilidad, este valor de adaptabilidad es proporcionado por la función objetivo. En este caso se considera la función objetivo como la suma de la Integral del Error Absoluto (IEA) definida en la ecuación (2).

$$IEA = \int_0^{\infty} abs[e(t)]dt \quad (2)$$

3.2 Implementación de la metodología en Software

En esta sección se describe la implementación de la metodología en software. El software utilizado para esta implementación fue MATLAB, en este se diseñaron de manera gradual distintos archivos script donde contaba con cada una de las partes de la

metodología. En el Listado 3.1 mostrado a continuación se presenta la descripción del código correspondiente a cada paso que se utilizó para la validación de la metodología en software.

Como primer punto se tiene la señal original con disturbio, la cual es obtenida mediante funciones en MATLAB, por lo que se especifica en el programa la duración de la señal, la duración y amplitud del disturbio, así como también se selecciona que tipo de disturbio se quiere generar.

Listado 3.1. Descripción de la metodología basada en software MATLAB para señales generadas artificialmente.

```

8      %% Señal Simulada
9      % Parametros de la señal artificial
10
11 -   sig_type = 2;           % Tipo de la señal (Disturbio) 1--> Sag ; 2--> Swell
12 -   f = 60;                % Frecuencia de la señal
13 -   Fs = 6000;            % Frecuencia de muestreo
14 -   Ts = 1/Fs;           % Periodo de muestreo
15 -   teta = 0;             % Desfasamiento
16 -   w = 2*pi*f;          % Argumento de la sinusoida
17 -   A = 127;             % Amplitud de la señal
18
19 -   ti = 0;                % Tiempo inicial
20 -   tf = 0.4;             % Tiempo final
21 -   t = ti : Ts : tf-Ts; % Vector de tiempo
22 -   T = tf;                %
23 -   dB = 20;              %
24 -   noise = 0;           % Ruido
25 -   t1 = 0.12;           % Tiempo inicio del disturbio
26 -   t2 = 0.26;           % Tiempo final del disturbio
27 -   alfa = 0.2;          % Porcentaje del disturbio
28
29      % Generacion de la señal artificial
30 -   vsig = PQdisturbance(sig_type, alfa, t1, t2, noise, dB, T, Fs, t, A, f, teta);
31 -   vsig = vsig';
32
33 -   figure('name', 'Señal de Analisis')
34 -   plot(t, vsig)
35 -   title('Señal original con disturbio');
36 -   xlabel('Tiempo [s]');
37 -   ylabel('Amplitud [V]');
38

```

Otra forma de obtener la señal es cargando un archivo .txt en MATLAB donde dicho archivo contiene una señal muestreada con un disturbio eléctrico, esto fue adquirido de una base de datos de la IEEE y algunas otras de pruebas experimentales obtenidas por el grupo HSP Digital, (Gutiérrez, 2014). El Listado 3.2 muestra la descripción descrita previamente.

Listado 3.2. Descripción de la metodología basada en software MATLAB para señales obtenidas a partir de base de datos.

```
8 %% Señal de base de datos
9 % Parametros de la señal
10 - f = 60;
11 - w = 2*pi*f;
12 - Fin = 7680;
13 - Fs = 6000;
14 - d = load('SWEL1.txt');
15 - longitud = length(d);
16 - idx = 0:longitud-1;
17 - di = [idx' d];
18 - t1 = 0:1/Fs: (longitud*(1/Fs))-(1/Fs);
19 - max_val =max(d);
20 - norm = d/max_val;
21
22 - [vsig, t] = Remuestrea(d,Fin,Fs);
23
24 - figure('name','Señal de análisis')
25 - plot(t,vsig)
26 - title('Señal original con disturbio');
27 - xlabel('Tiempo [s]');
28 - ylabel('Amplitud [p.u.]');
29
```

Una vez que se tiene la señal con disturbio, se aplica un detector de cruce, descrito en el Listado 3.3, esto para seccionar la señal en medios ciclos, y así aplicar el método de Micro-Algoritmos Genéticos, ya que por cada medio ciclo se realizan las iteraciones correspondientes.

Listado 3.3. Descripción del detector de cruce por cero.

```
39 %% Cruce por cero
40 |
41 - [Idx_CC] = CCidx(vsig)
42
```

Después se aplica a la señal un filtro pasabandas y un rechazabandas, con la finalidad de obtener la señal a 60 Hz que es a la frecuencia estándar que la red eléctrica tiene en México, con el filtro pasabandas se obtiene la señal deseada de 60 Hz, y con el filtro rechazabandas se obtienen todas las demás señales que pueden estar en la señal original y que de alguna manera podrían afectar al momento de aplicar la metodología, señales como ruido o alguna otra afectación a la señal. El Listado 3.4 describe los dos tipos de filtros utilizados en la metodología en software MATLAB.

Listado 3.4. Descripción de los filtros pasabandas y rechazabandas.

```

42     %% Señal Fundamental y Remanentes
43
44     % Filtro Pasabandas
45 -   PB_filt = Filtro_PBandas;
46     %
47     % Filtro Rechazabandas
48 -   RB_filt = Filtro_RBandas;
49     %
50     % Extraccion de la señal fundamental
51 -   ym = filter(PB_filt,vsig);
52
53     %
54     % Extraccion de las señales remanentes
55 -   yr = filter(RB_filt,vsig);
56     %
57 -   figure('name','Señal Fundamental')
58 -   plot(t,ym,'g');
59 -   title('Señal Fundamental');
60 -   xlabel('Tiempo [s]');
61 -   ylabel('Amplitud [V]');
62     %
63 -   figure('name','Señales Remanentes')
64 -   plot(t,yr,'m');
65 -   title('Señales Remanentes');
66 -   xlabel('Tiempo [s]');
67 -   ylabel('Amplitud [V]');
68

```

Ya que se tenía la señal original, se aplica un detector de fase, porque en algunas señales no comienzan desde cero o se encuentran desfasadas respecto a una señal sinusoidal pura que comienza en cero, esto es porque el método de Micro-Algoritmos Genéticos se aplica a cada medio ciclo y por lo tanto la señal no debe de estar desfasada ya que esto puede proporcionar resultados incorrectos. El Listado 3.5 describe el detector de fase descrito anteriormente.

Listado 3.5. Descripción del detector de fase

```

70     %% Detector de Fase
71
72 -   [ph] = psd_s(t, ym, f);     % Detector de fase

```

Una vez realizado lo anterior se aplica la metodología de los Micro-Algoritmos Genéticos, para esto se definen algunos parámetros como el número de generaciones que se desean, y la cantidad de individuos, en este trabajo se establece el número de generaciones

en 30 y el número de individuos en 4. En el Listado 3.6 se describen los parámetros de la presente metodología en software MATLAB.

Listado 3.6. Descripción de los parámetros para nuestra metodología en software MATLAB

```

72  %% Parametros Genereales del MAG
73
74 - G    = 30;           % Generaciones
75 - Ps   = 4;           % Poblacion
76 - Mp   = 0.2;        % Probabilidad de mutacion
77 - Am   = 127;        % Amplitud de la señal
78 - Rng  = [0.0 2*Am]; % Rango de la variable de diseño
79
80  % *** ----- Analisis de CE en la señal fundamental con MAG ----- ***
81  % *** ----- *** %
82 - Ftns  = zeros(Ps,1);
83 - NPop  = zeros(Ps,1);
84 - best  = zeros(G,1);
85 - ovalue = zeros(G,1);
86  %
87  % Total de medios ciclos para analizar|
88 - HalfCycles = length(czidx);
89 - Ae_b      = zeros(HalfCycles,1);
90  %
91 - c = 2;
92 - ini = 1;
93 - fin = czidx(1);

```

El proceso empieza con 4 individuos escogidos aleatoriamente entre el rango de 0 a 2, los individuos se evalúan en la ecuación 1, cada individuo representa la amplitud de la señal, por cada individuo se tiene un resultado, el cual se evalúa en función del error respecto a la señal original.

El individuo que proporciona el menor error es nuestra mejor solución, y los siguientes dos mejores se colocan debajo del mejor, el cuarto individuo es el que nos proporciona mayor error y por lo tanto no es la mejor solución.

Para la siguiente generación el mejor individuo se conserva, ya que al ser nuestra mejor solución no tiene sentido eliminarlo, los siguientes dos mejores se utilizan para hacer la operación de cruce, donde se toman partes de cada individuo y junto con el primer individuo, que obtuvo la mejor solución, se genera un nuevo individuo, de esta manera estos dos individuos se convierten en los padres de posibles nuevas soluciones para el problema, también se tiene una probabilidad de mutación, la cual es baja, pero si llega a

ocurrir, los individuos resultantes de la cruce se modifican, dando resultado a posibles nuevas soluciones al problema, por último, el cuarto individuo que fue la peor solución se sustituye por otro individuo al azar dentro del espacio de soluciones, esto proporciona al método mayor variabilidad.

El proceso se repite el número de iteraciones que se deseen, cada iteración es una nueva generación de individuos, en esta caso se realizan 30 iteraciones, en las que se obtenía un buen resultado al finalizar las iteraciones. El Listado 3.7 describe el proceso iterativo de la metodología en software MATLAB.

Listado 3.7. Proceso iterativo del Micro-Algoritmo Genético.

```

94      %Proceso Iterativo
95      while( c <= HalfCycles + 1)
96      %   Poblacion Inicial Aleatoria
97      Pop = Rng(1,1) + ( Rng(1,2) - Rng(1,1) ) * rand(Ps,1);
98
99      % Auxiliar time vector and signal vector (Half-cycle)
100     tx = t(ini:fin);
101     yx = ym(ini:fin);
102
103     iter = 1;
104     while(iter <= G)
105         %Evaluacion de la Población
106         IEA = 0;
107         for j = 1 : Ps
108             Ae = Pop(j,1);
109             ye = Ae*cos(w*tx + ph)';
110             e = yx-ye;
111             IAE = sum(abs(e));
112             Ftns(j,1)=IAE;
113         end
114
115         %Ordenamiento de la Poblacion de acuerdo a su valor de adaptabilidad
116         for j = 1 : Ps
117             for k = 1 :Ps - 1
118                 if Ftns(k) >= Ftns(k+1)
119                     aux = Ftns(k);
120                     Ftns(k)=Ftns(k+1);
121                     Ftns(k+1)=aux;
122
123                     aux=Pop(k,1);
124                     Pop(k,1)=Pop(k+1,1);
125                     Pop(k+1,1)=aux;
126                 end
127             end
128         end

```



```

129     %Operacion Cruza
130     NPop(1,1) = Pop(1,1);
131     for z = 2 : Ps - 1
132         NPop(z,1) = ( Pop(1,1) + Pop(z,1) ) / 2;
133     end
134     NPop(Ps,1) = Rng(1,1) + (Rng(1,2) - Rng(1,1))*rand(1,1);
135     %Operacion Mutacion
136     for j = 2 : Ps - 1
137         if rand < Mp
138             NPop(j,1) = Rng(1,1) + (Rng(1,2) - Rng(1,1))*rand(1,1);
139         end
140     end
141     Pop = NPop;
142     best(iter)=Pop(1,1);
143     ovalue(iter)=Ftns(1,1);
144     iter = iter + 1;
145 end
146
147 Ae_b(c-1) = best(iter-1);
148 Ae_f(ini:fin)=best(iter-1);
149
150 if c <= HalfCycles
151     ini = czidx(c-1);
152     fin = czidx(c);
153 end
154 c = c + 1;
155 end
156
157 figure('name','Analiis en la señal fundamental');
158 plot(ym,'g');
159 hold on
160 plot(Ae_f,'k.-');
161 title('Analiis en la señal fundamental');
162 ylabel('Amplitud [V]');
163 xlabel('Tiempo [s]');
164 legend('Señal Fundamental (Pasabanda)','Señal Envolvente','location','best');
165
166 figure('name','Señal Envolvente en la señal fundamental');
167 plot(Ae_b,'m--');
168 title('Señal Envolvente en la señal fundamental');
169 ylabel('Amplitud [V]');
170 xlabel('Tiempo [s]');
171

```

Con este proceso se validó la metodología en software lo que permitió seguir a delante con el proceso descrito en la metodología.

3.3 Diseño de las arquitecturas para la implementación en Hardware

Para la implementación en hardware de la metodología propuesta se consideró el diagrama mostrado en la Figura 3.2. Se observa que la metodología no se encuentra como una función en software, es simplemente el esquema de los MAG con una estructura genérica tal que permite hacer la descripción de una arquitectura digital. La estructura del diagrama tiene como entradas y salidas datos digitales y dentro del bloque se tienen las operaciones relacionadas con eventos genéticos. En los siguientes párrafos se describe de manera detallada la implementación en hardware.

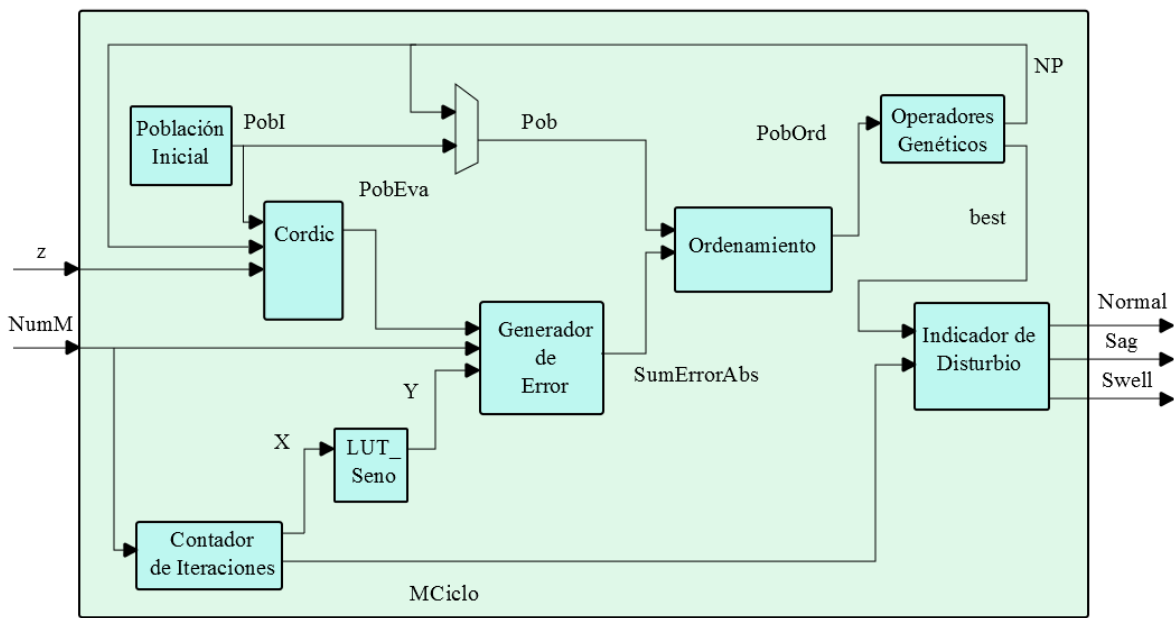


Figura 3.2. Diseño de las arquitecturas digitales para la implementación en Hardware

La Figura 3.2 consta de varios bloques donde cada uno tiene una función específica dentro del sistema y que forma parte de la metodología seguida de MAG, en los párrafos siguientes se muestran y describen a detalle cada uno de estos módulos.

El primer bloque, mostrado en la Figura 3.3 es el de la población inicial, en este bloque se elige la población de 4 individuos entre un rango de 0 a 2, para este proceso se

generaron los individuos aleatoriamente mediante código en C, se almacenaron en un archivo txt y el módulo en vhdl se encarga de obtenerlos del archivo y convertirlos señal.

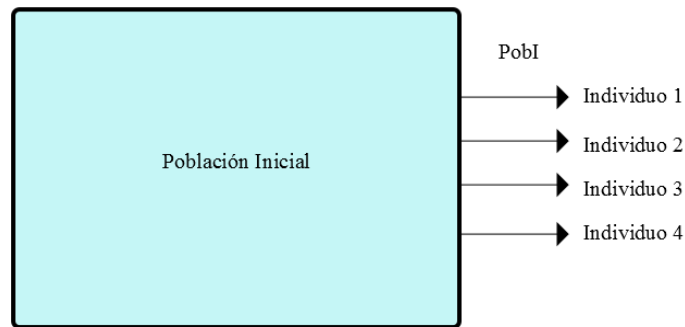


Figura 3.3. Bloque Población Inicial de las arquitecturas en VHDL.

El Bloque Cordic mostrado en la Figura 3.4 se utilizó para evaluar la población, se utilizó el algoritmo CORDIC debido a que permite representar una onda sinusoidal de manera muy aproximada, esto sin requerir demasiados recursos computacionales , en este bloque la población inicial o la nueva población multiplica la respuesta del algoritmo CORDIC que para éste caso es un punto de la señal sinusoidal que representa la forma de onda eléctrica y de esta manera obtener una población evaluada que es la magnitud de la señal sinusoidal correspondiente a cada uno de los individuos.

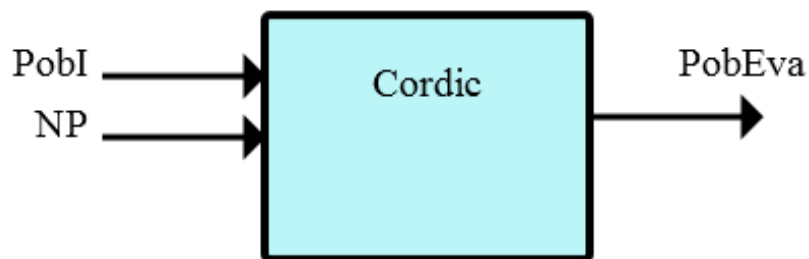


Figura 3.4. Bloque Cordic de las arquitecturas en VHDL.

El bloque LUT_seno es una look up table donde se encuentran los valores de la señal eléctrica medida, dentro de esta se encuentra el disturbio eléctrico ya sea un sag o un swell, para cada valor de X nos entrega un valor de Y que es el valor de la señal sinusoidal en el instante de tiempo muestreado X . El bloque se muestra en la Figura 3.5.

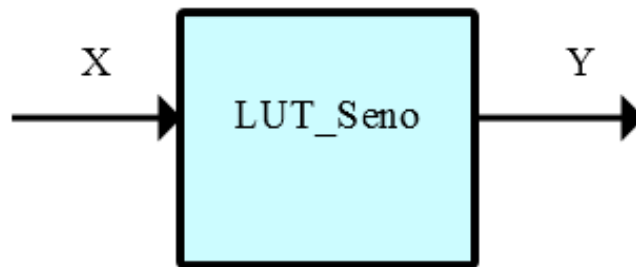


Figura 3.5. Bloque LUT_Seno de las arquitecturas en VHDL.

El bloque generador de error mostrado en la Figura 3.6 toma el valor de los dos bloques anteriores descritos y los compara, es decir toma el valor de la señal Y del bloque lut_seno y lo resta al valor obtenido del cordic, la referencia menos el valor obtenido con la población, este valor es el error, como la función objetivo es la sumatoria del error absoluto, durante medio ciclo se obtienen los valores del error en cada punto de muestreo y se van sumando, de esta manera al terminar el medio ciclo se obtiene para cada individuo un valor de fitness el cual es la suma de los errores absolutos en cada punto de muestreo.

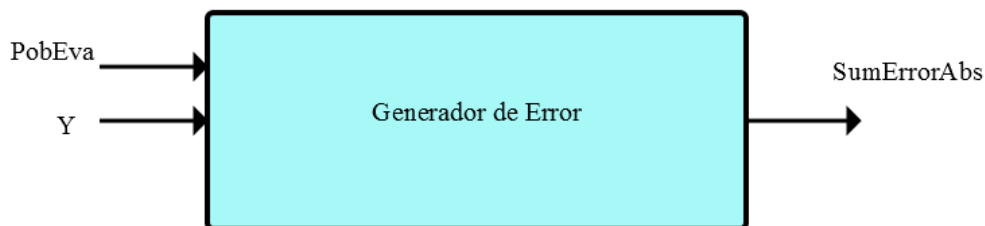


Figura 3.6. Bloque Generador de Error de las arquitecturas en VHDL.

El bloque ordenamiento que se muestra en la Figura 3.7 ordena el valor de fitness de cada individuo de manera que el individuo que tenga menor error este al principio del vector, ordena tanto el valor del error absoluto de la población de tal manera que correspondan los valores del error con cada individuo.

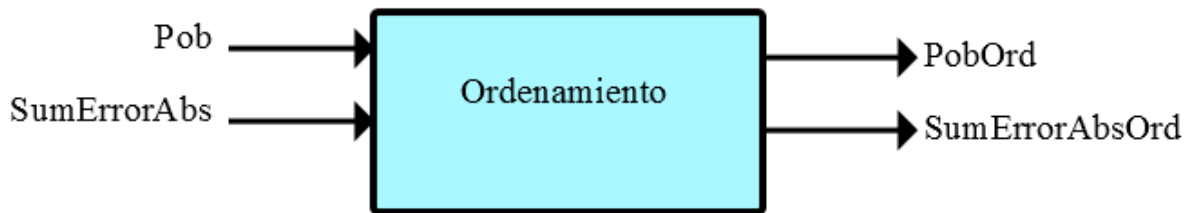


Figura 3.7. Bloque Ordenamiento de las arquitecturas en VHDL.

El bloque de operadores genéticos realiza las operaciones de cruce y mutación para obtener la nueva población, primero se mantiene el mejor individuo que se obtuvo del ordenamiento, los siguientes dos individuos se crean a partir del mejor individuo y del segundo y tercer individuo mediante la cruce, esto es sumando los 2 individuos y dividiéndoles entre 2 de tal forma que se tiene la mitad del mejor individuo junto con la mitad del segundo y tercer individuo para crear dos individuos nuevos que proporcionen un mejor desempeño en la función objetivo, el último individuo es seleccionado aleatoriamente del espacio de diseño, esto con la intención de darle mayor variabilidad al método, este individuo es obtenido desde un archivo txt que fue previamente generado con el número de individuos aleatorios que se utilizaron, es decir el número de iteraciones que se realizaron del método, en este caso de estudio el valor fue de 30. Este bloque se muestra en la Figura 3.8.

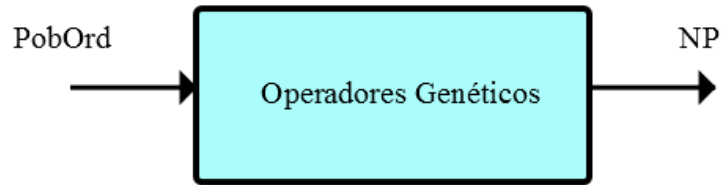


Figura 3.8. Bloque Operadores Genéticos de las arquitecturas en VHDL.

El bloque contador de iteraciones determina el número de ciclos que van pasando así como las iteraciones del micro algoritmo genético, las entradas para este bloque son las señales de clk que es una señal de reloj utilizada en el sistema, y la señal H que es la cantidad de muestras en las que esta seccionados los medios ciclos. El bloque de contador de iteraciones se muestra en la Figura 3.9.

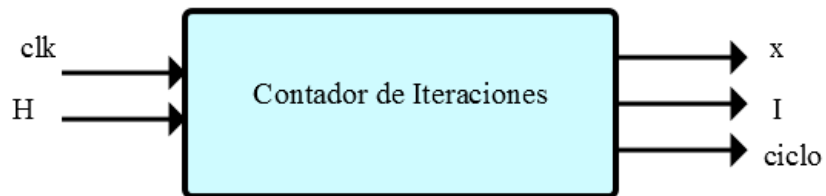


Figura 3.9. Bloque Contador de Iteraciones de las arquitecturas en VHDL.

El bloque indicador de disturbios mostrado en la Figura 3.10 determina al final de cada medio ciclo analizado si existió un cambio en la amplitud de la señal y por lo tanto determina si existió un disturbio sag o swell, o incluso si no existió ningún cambio esto se indica en una señal que tiene como nombre 'normal'.

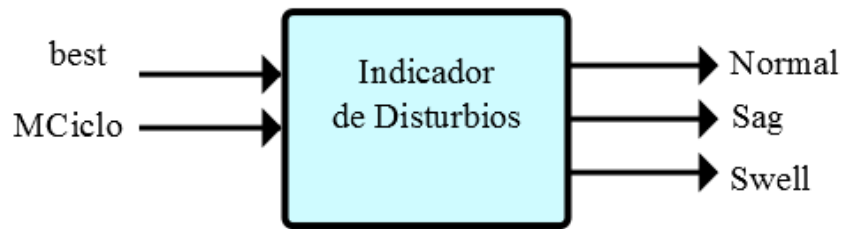


Figura 3.10. Bloque Indicador de Disturbios de las arquitecturas en VHDL.

El bloque más jerárquico mostrado en la Figura 3.11 tiene como salidas 3 señales, sag, swell y normal las cuales se activan dependiendo el valor de salida que tenga el mejor individuo al término de cada proceso de nuestro micro algoritmo genético.

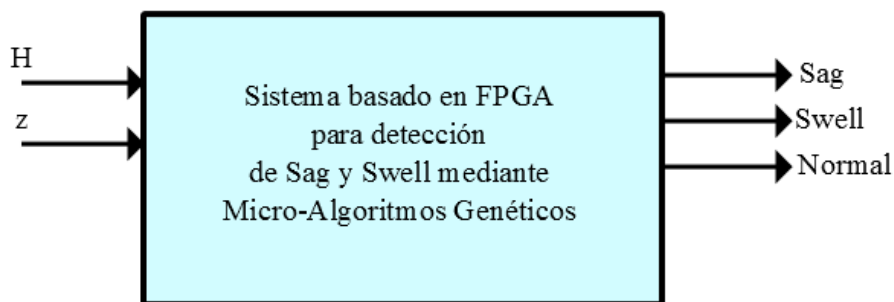


Figura 3.11. Bloque Jerárquico de las arquitecturas en VHDL

3.4 Plataforma de pruebas

Para poder validar y verificar la metodología propuesta es necesario contar con una plataforma de pruebas donde se podrá implementar la metodología. Se cuenta con una máquina fresadora CNC de tres ejes en la cual se le inducirán los disturbios eléctricos a través de un banco de pruebas en el cual se podrán generar los disturbios. El diagrama de

conexión del banco de pruebas se muestra en la Figura 3.12. Se puede notar en la figura los elementos que lo componen y su descripción está dada a continuación.

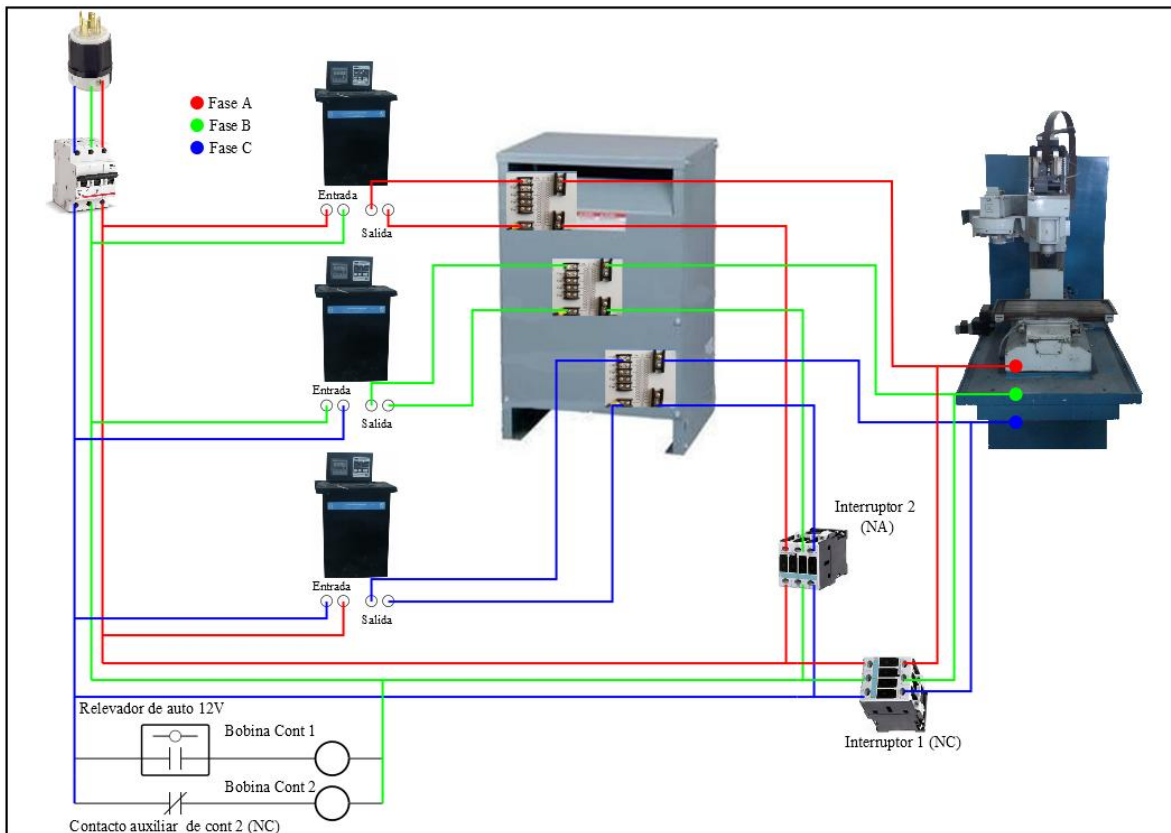


Figura 3.12. Diagrama de conexión del banco de pruebas (generación de disturbios)

El diagrama del generador de disturbios eléctrico Sag y Swell que alimenta la máquina fresadora CNC está representado por los componentes para las tres fases de la línea de energía eléctrica. Las variaciones en la tensión de entrada son posibles a través de una disposición de un transformador variable, un acoplamiento a un auto-transformador y un relevador de estado sólido (SSR). El transformador variable permite la regulación de la tensión suministrada por el auto-transformador de acoplamiento. La disposición de los contactos normalmente abiertos y cerrados junto con el SSR permite la selección de la tensión que alimenta la Fresadora CNC. Cuando el SSR no está activado, la máquina se alimenta con la tensión nominal de la línea eléctrica. Cuando se activa el SSR, el contacto NC está abierto y el contacto NA se cierra. En este caso, el voltaje seleccionado se

resta del voltaje nominal, induciendo el hueco de tensión (Vs SAG) en la línea eléctrica que alimenta el torno CNC. El generador SAG tiene la capacidad de inducir huecos de tensión momentáneas durante la operación de torno CNC. El rango del generador de hundimiento es de 50% a 100% de la tensión de fase con la corriente nominal de 30 A.

Para hacer la adquisición de las señales de voltaje generadas por el sistema propietario de generación de disturbios eléctricos se utilizaron las tarjetas PLCUAQ y una tarjeta de adquisición de datos o sistema de monitoreo, en la Figura 3.13 se muestra el sistema completo de generación de disturbios con el de adquisición de datos.

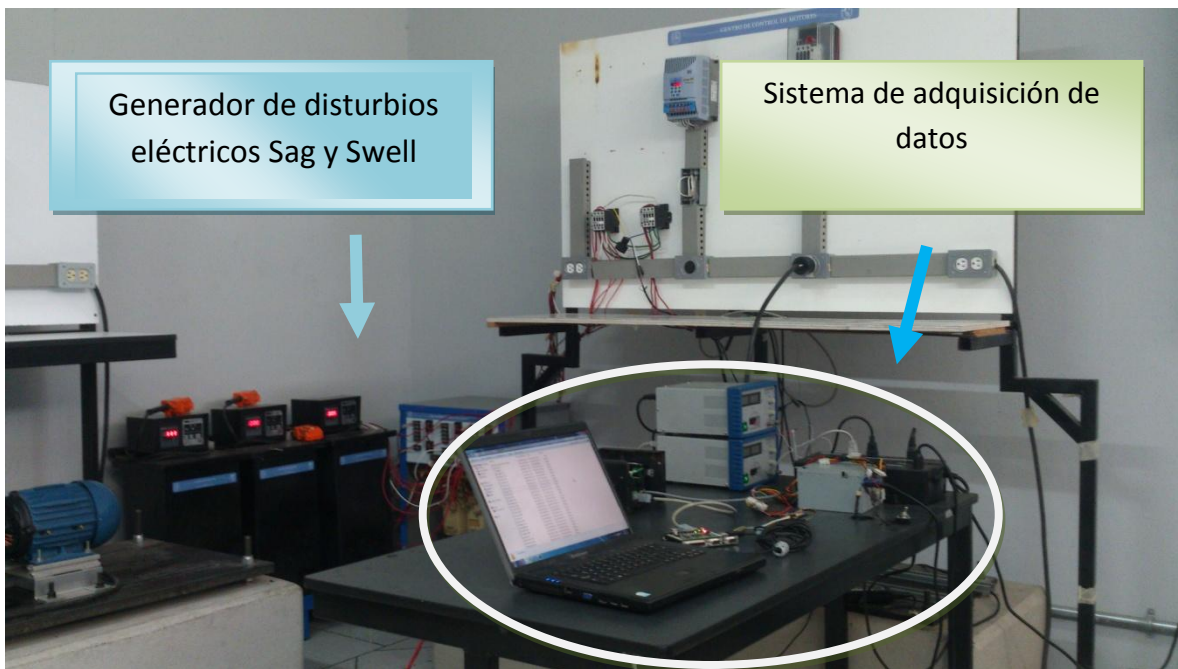


Figura 3.13. Sistema propietario de generación de disturbios eléctricos sag y swell junto con el sistema de adquisición de datos.

En la Figura 3.14 se muestra de manera más específica el sistema de adquisición de datos.

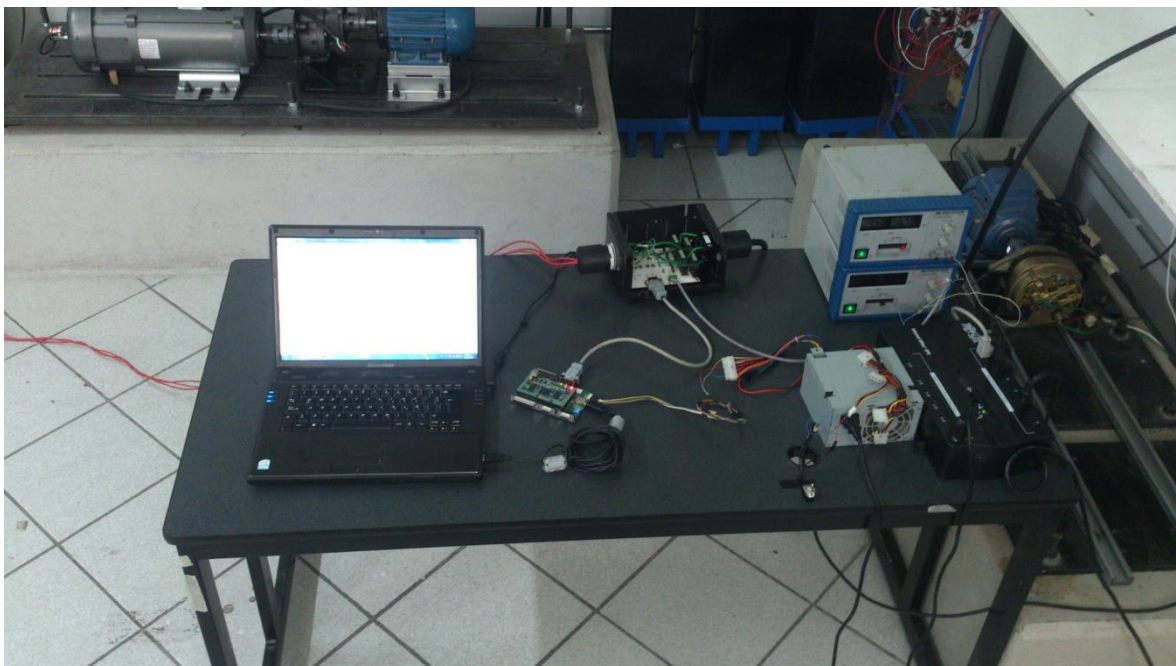


Figura 3.14. Sistema de adquisición de datos

El sistema de generación de disturbios estaba conectado a la máquina fresadora mostrada en la Figura 3.15, en la cual se llevaron a cabo las pruebas, por cuestiones de disponibilidad se realizaron las pruebas sobre el eje 'x' de la máquina el cual está indicado en la figura.



Figura 3.15. Máquina fresadora donde se realizaron las pruebas

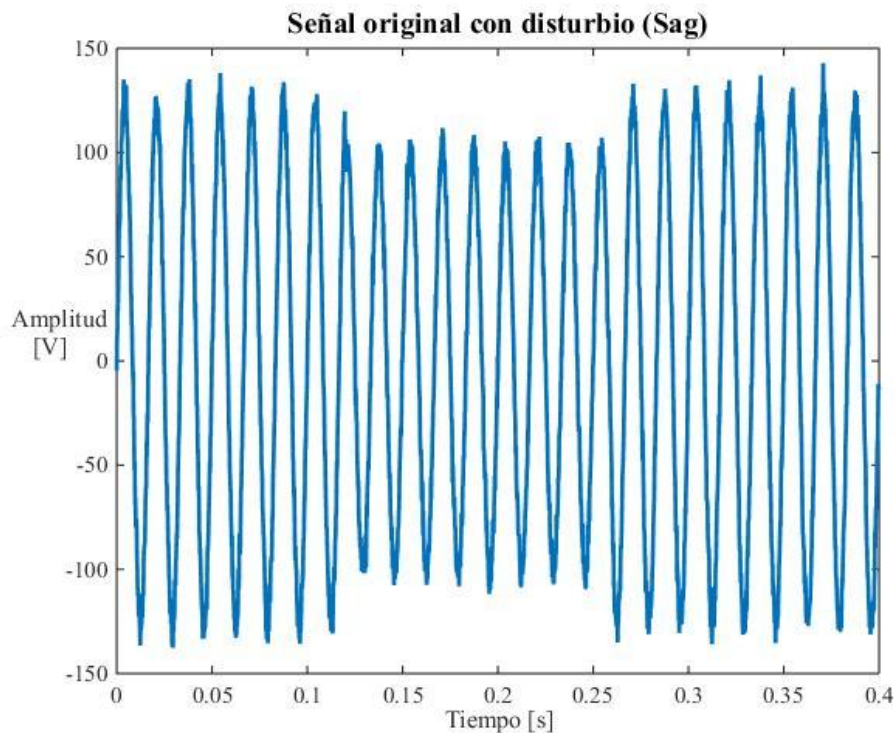
Capítulo 4

Resultados

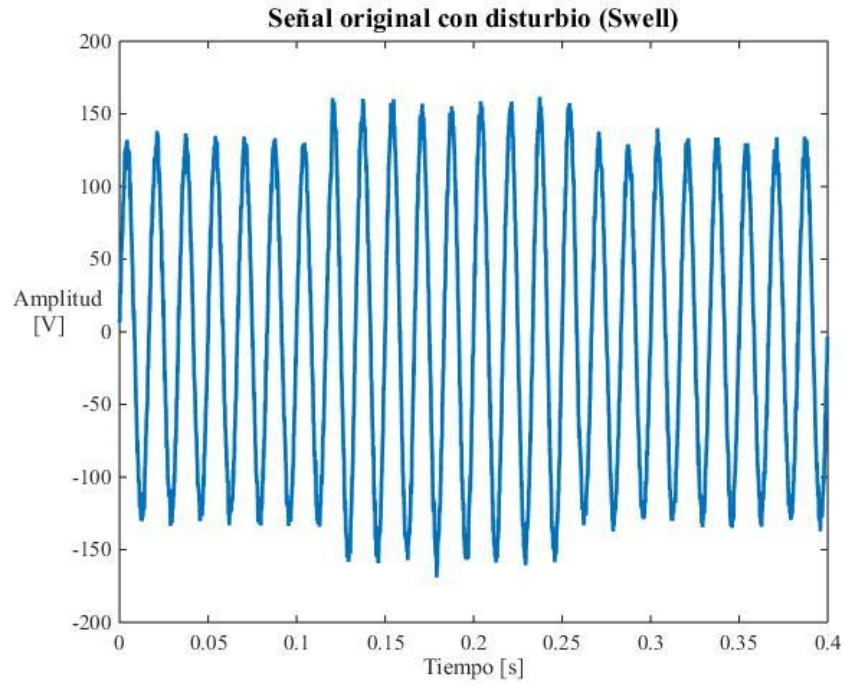
En este capítulo se presentan los resultados obtenidos para los distintos pasos dentro de la metodología seguida.

4.1 Resultados de la implementación en software

Primero se muestran las gráficas de la implementación en software donde se desarrollaron 2 casos de estudio, uno sobre señales sintéticas y otro para señales que fueron adquiridas desde una base de datos. La primera señal que se genera o se adquiere es la de la señal original con disturbio, para las señales simuladas o sintéticas se muestran los dos tipos de disturbios en la Figura 4.1, se muestra el sag y swell generados mediante funciones de MATLAB, a estas señales simuladas se les puede introducir ruido para afectar un poco la señal y observar si influye en el proceso de la metodología, en las señales mostradas en la Figura 4.1 se introdujo a las señales un ruido de 25 dB.



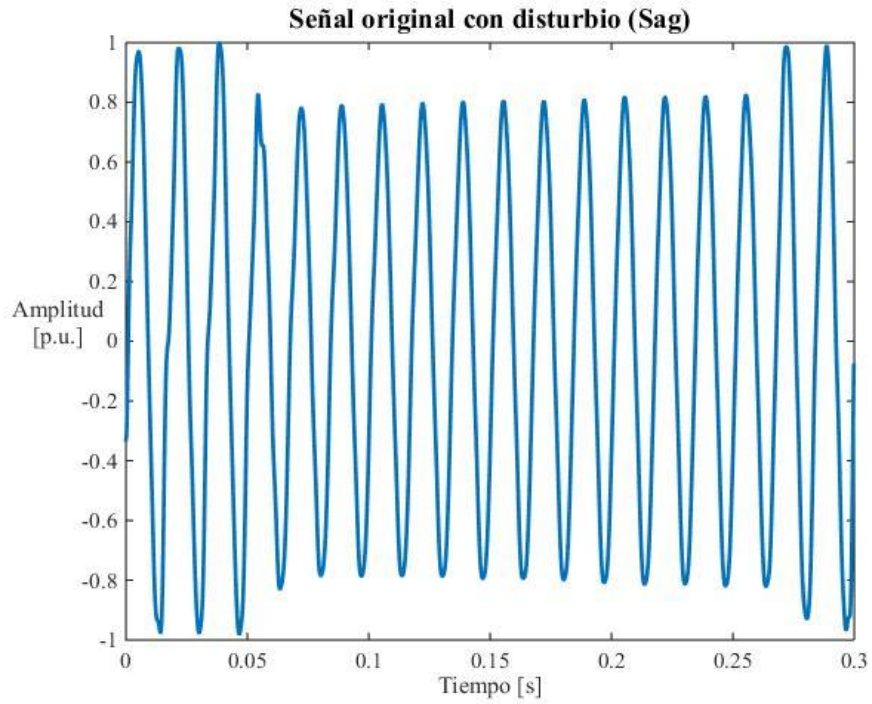
(a)



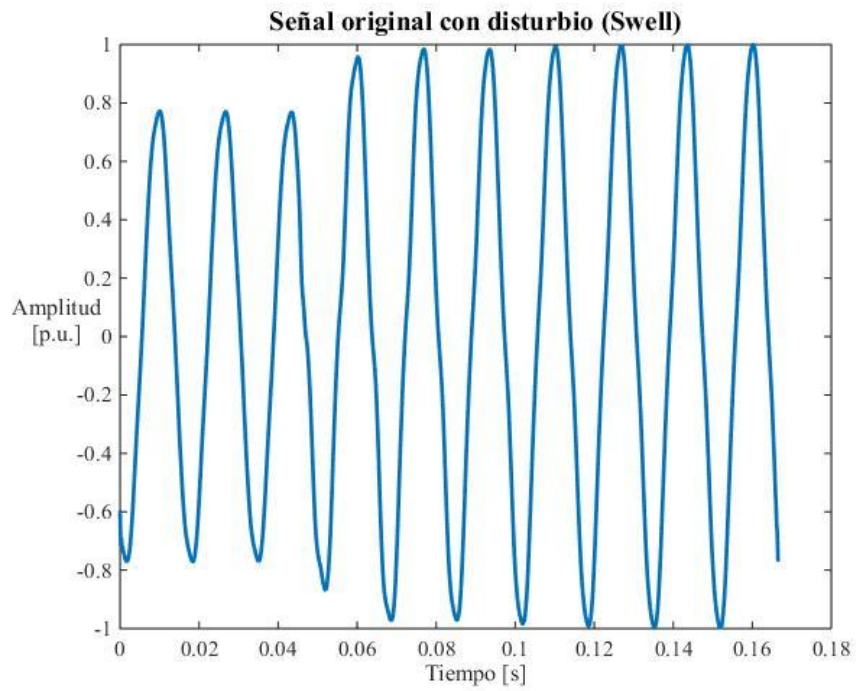
(b)

Figura 4.1. Señal Original con disturbio para señales sintéticas: (a) Sag y (b) Swell.

Para las señales que fueron cargadas desde una base de datos se muestran en la Figura 4.2.



(a)

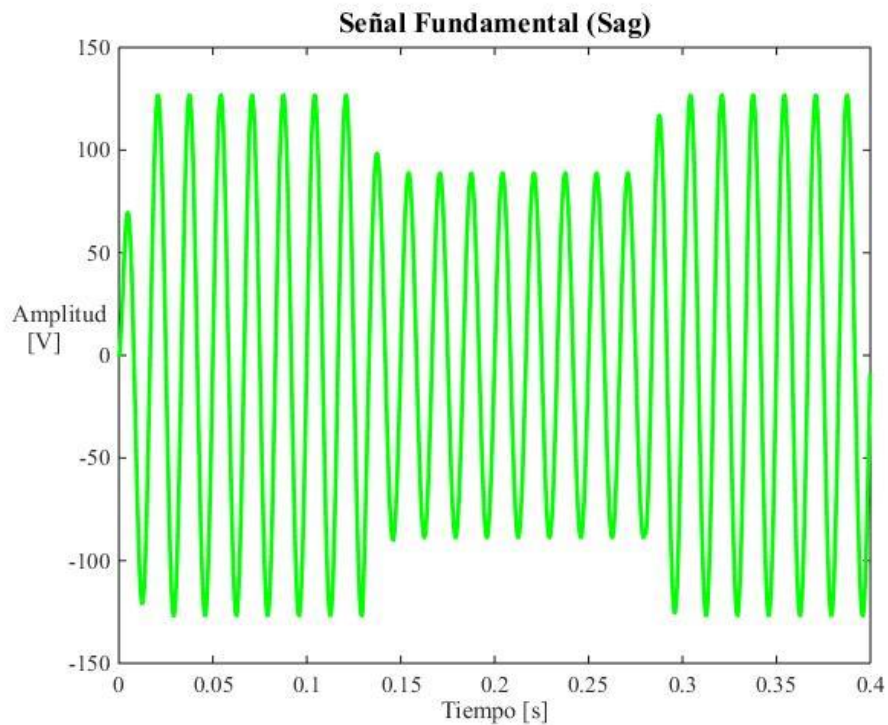


(b)

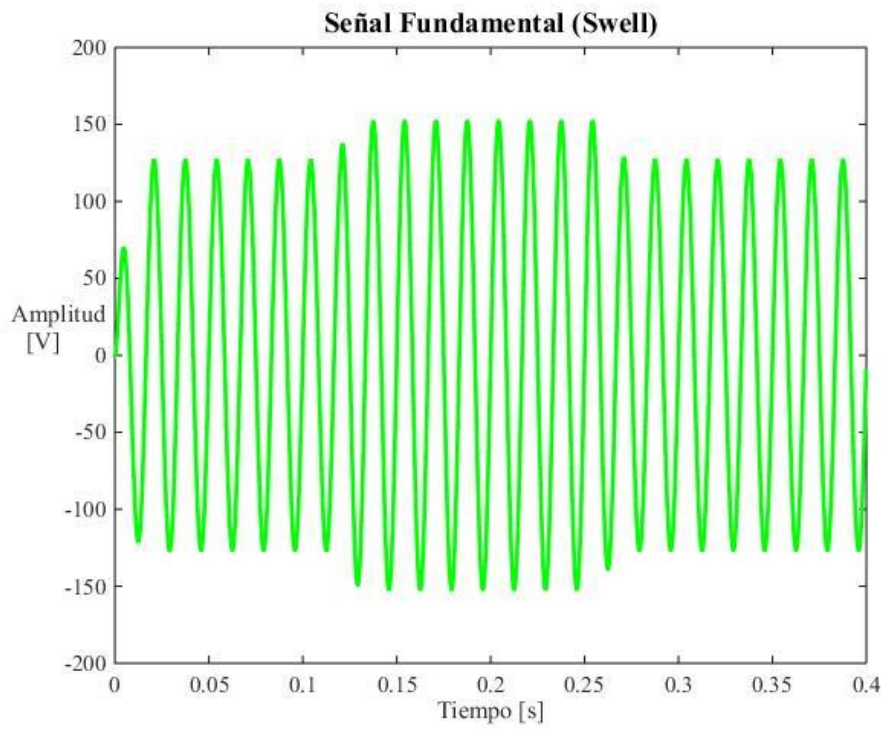
Figura 4.2. Señal Original con disturbio para señales de base de datos: (a) Sag y (b) Swell.

Después a esas señales, se les aplicó un filtro para obtener solamente la señal de 60 Hz que es la que interesa porque a esa frecuencia trabaja la red eléctrica en el país y además que la señal sintética o la de base de datos puede contener alguna otra anomalía que pudiera conducir a resultados incorrectos a la hora de realizar el análisis de MAG.

La Figura 4.3 muestra las señales fundamentales de las señales sintéticas y la Figura 4.4 muestra las señales fundamentales para las señales cargadas desde una base de datos.

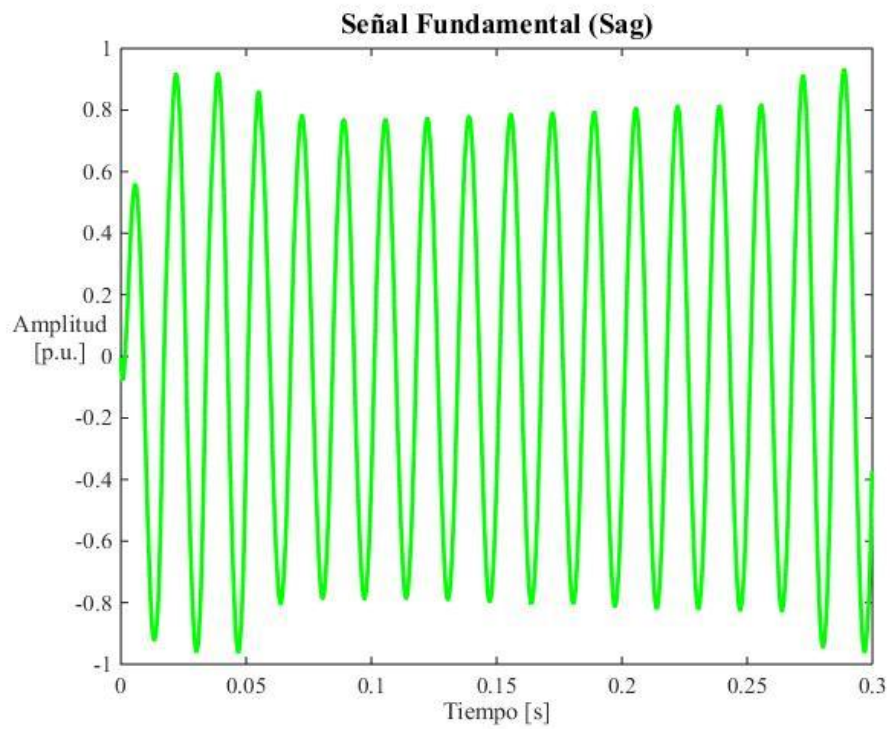


(a)

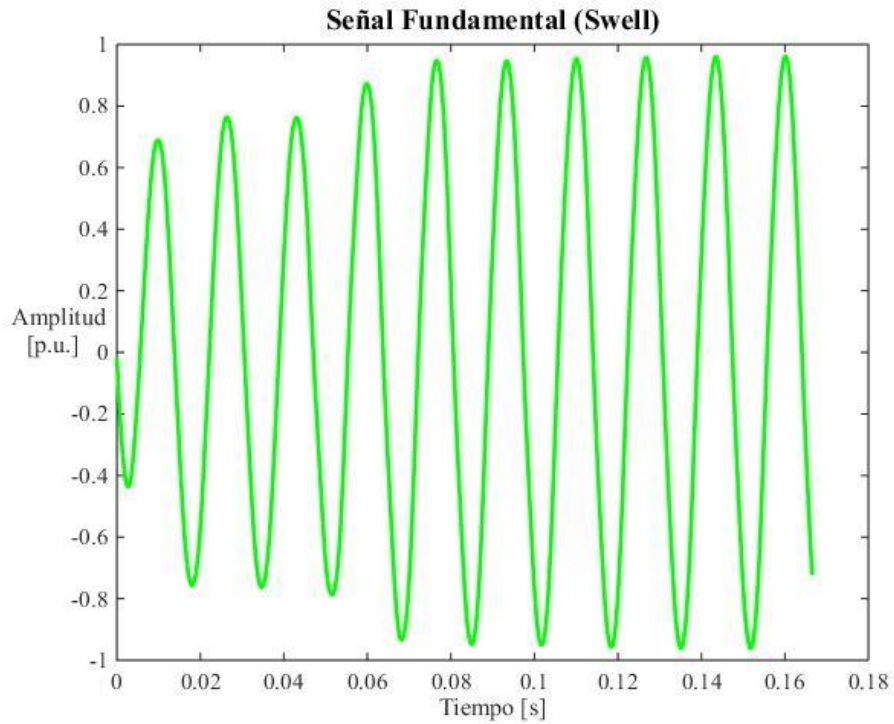


(b)

Figura 4.3. Señal fundamental para señales sintéticas: (a) Sag y (b) Swell.



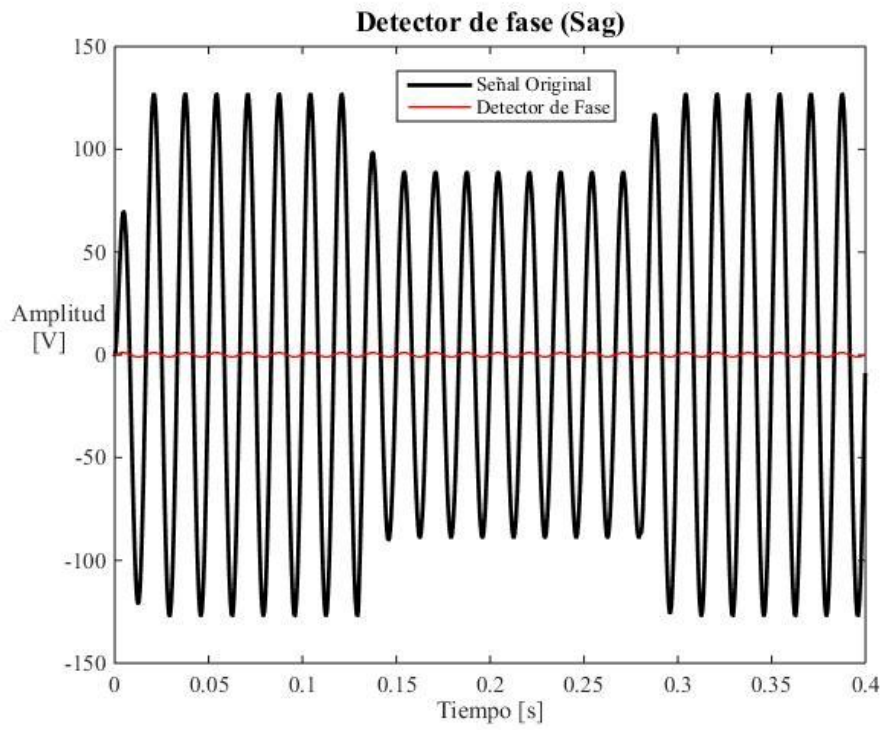
(a)



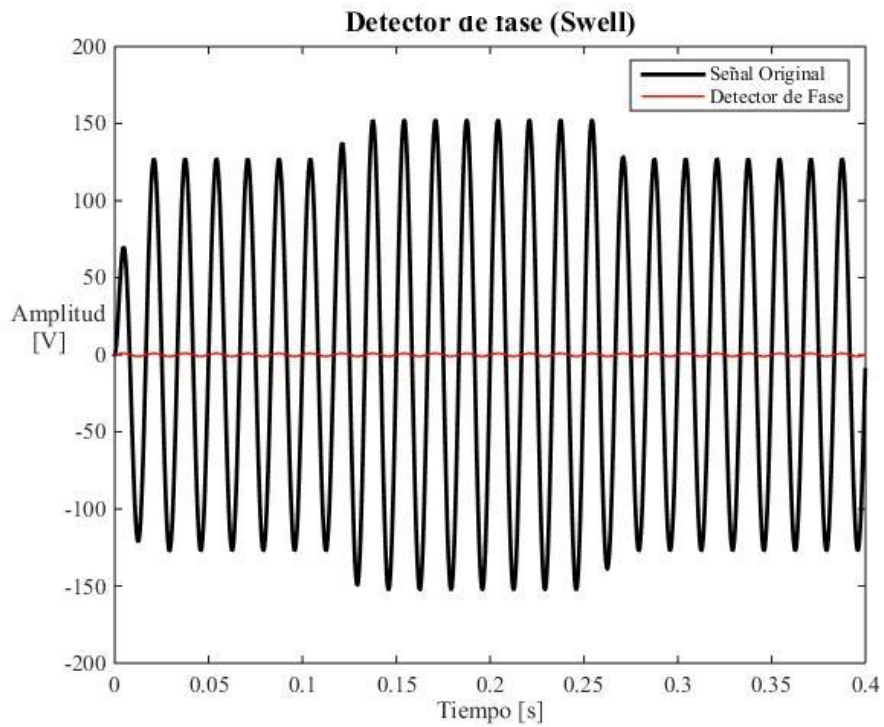
(b)

Figura 4.4. Señal fundamental para señales de base de datos: (a) Sag y (b) Swell.

Otro punto que es importante mencionar es que para el análisis de MAG se requiere un detector de fase, debido a que el proceso trabaja por cada medio ciclo de la señal eléctrica, y por lo tanto si no se considera desde el punto de inicio de un ciclo puede generar error el análisis, por esto se implementó un detector de fase, en la Figura 4.5 se muestra el detector de fase para las señales sintéticas y en la Figura 4.6 se muestra para las señales de base de datos.

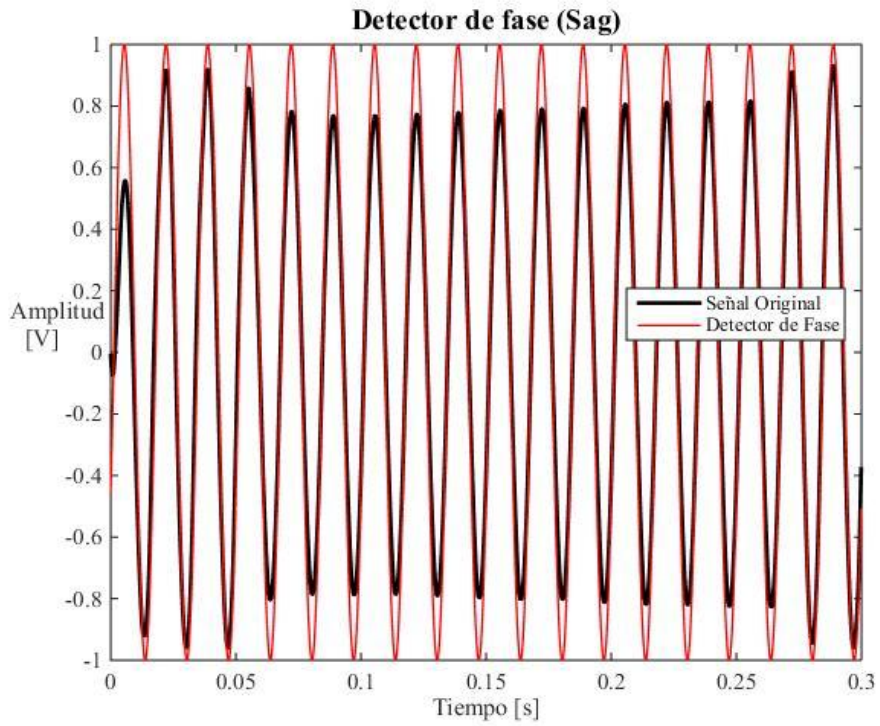


(a)

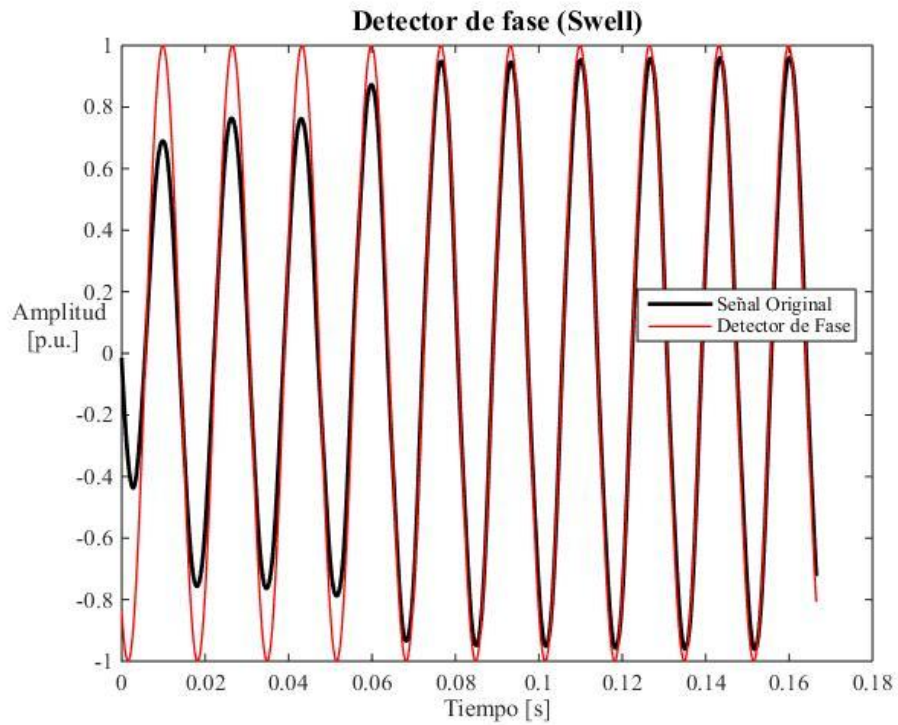


(b)

Figura 4.5. Detector de fase para señales sintéticas: (a) Sag y (b) Swell.



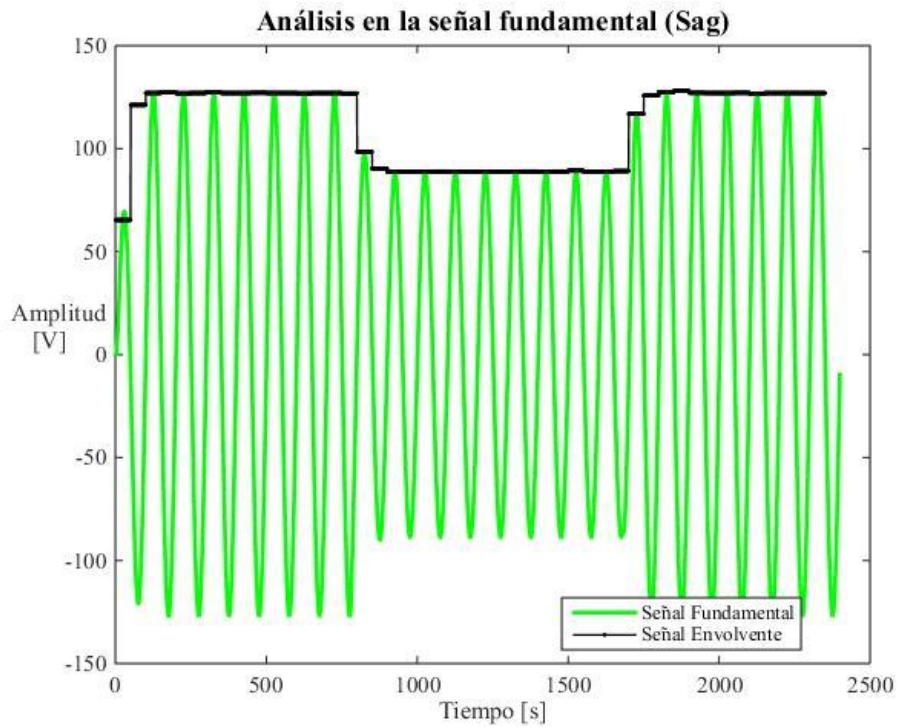
(a)



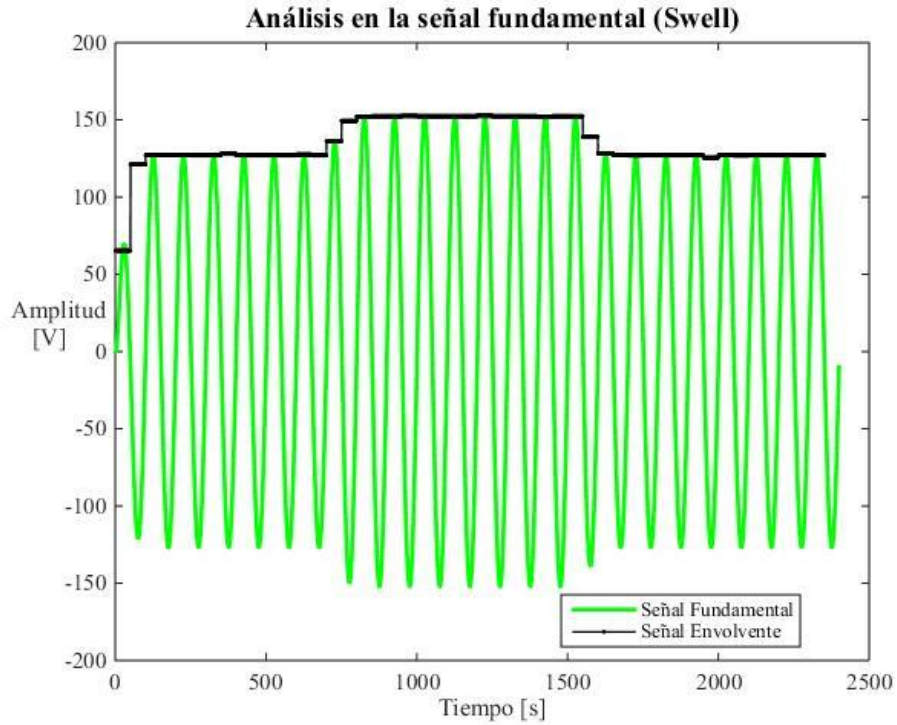
(b)

Figura 4.6. Detector de fase para señales de base de datos: (a) Sag y (b) Swell.

A la señal fundamental se le aplicó la metodología de micro-algoritmo genético para en cada medio ciclo obtener el máximo valor de esa señal y así ir creando una señal envolvente la cual indica la amplitud de la señal por cada medio ciclo. En la Figura 4.7 y la Figura 4.8 se muestran estas señales con su respectivos análisis de MAG.

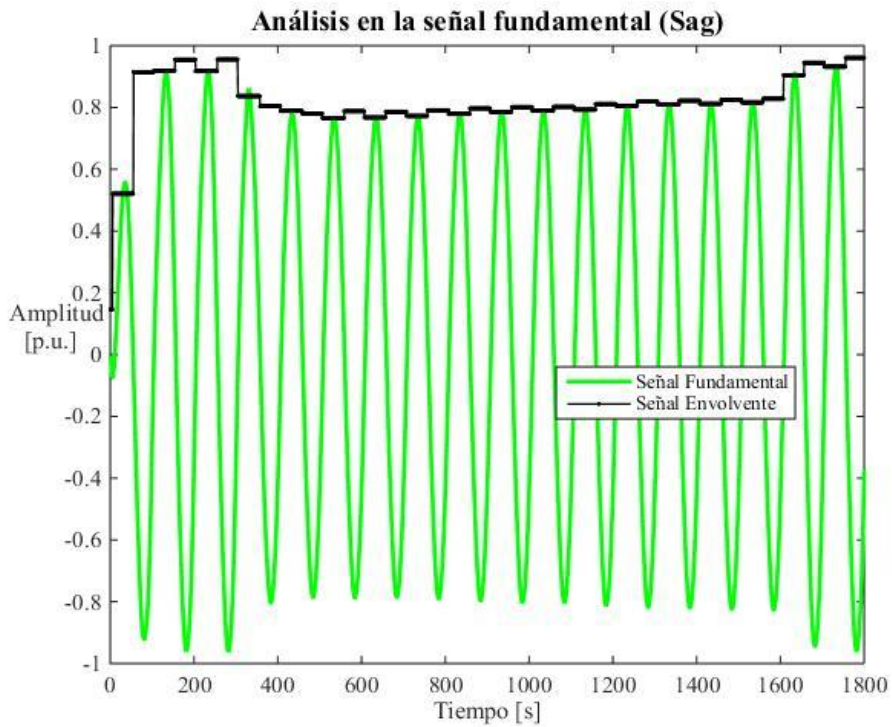


(a)

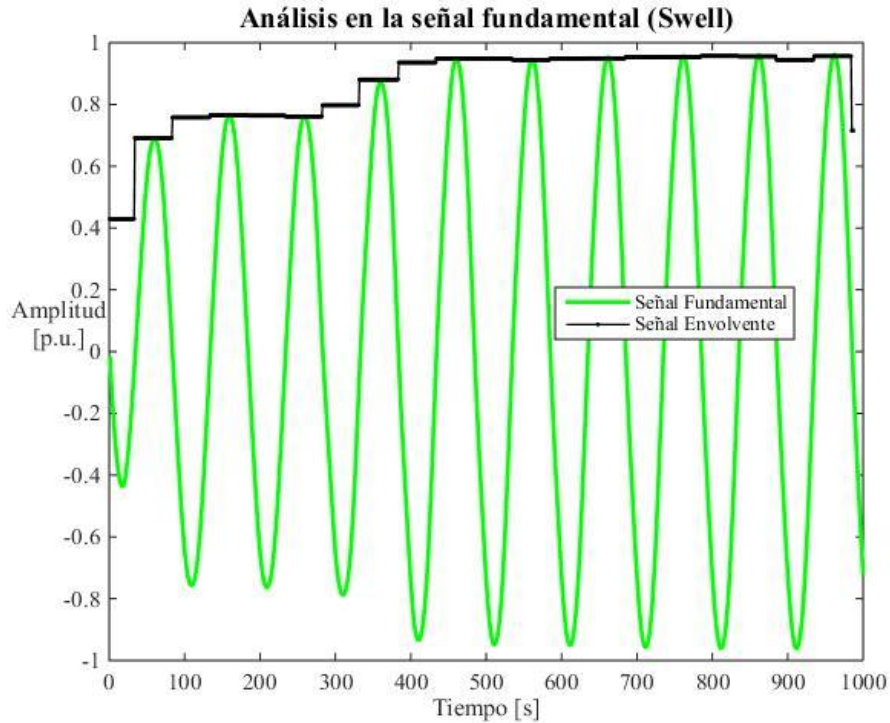


(b)

Figura 4.7. Análisis en la señal fundamental para señales sintéticas: (a) Sag y (b) Swell.



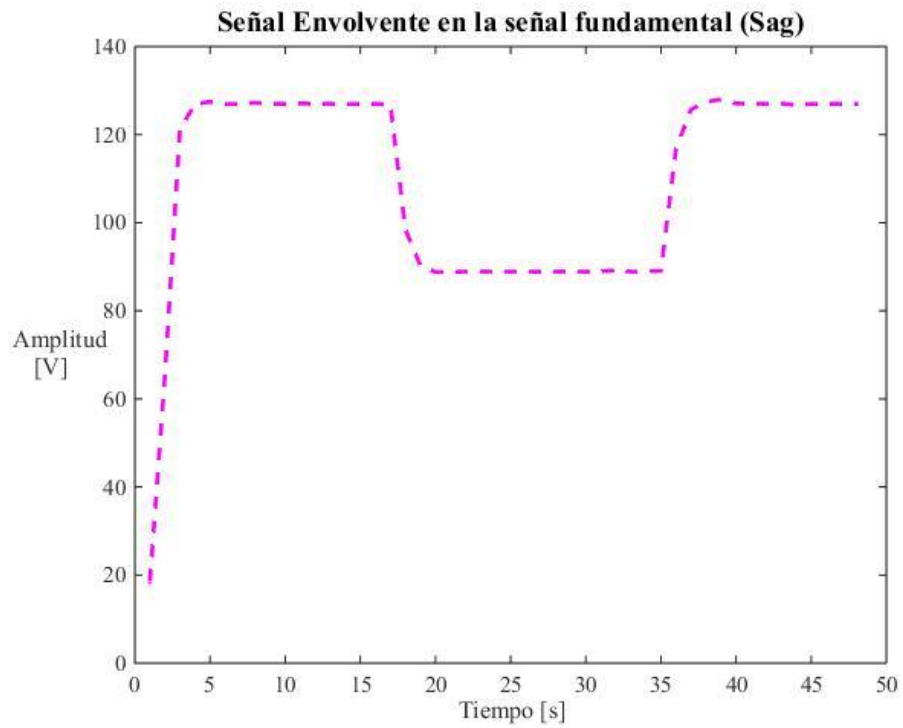
(a)



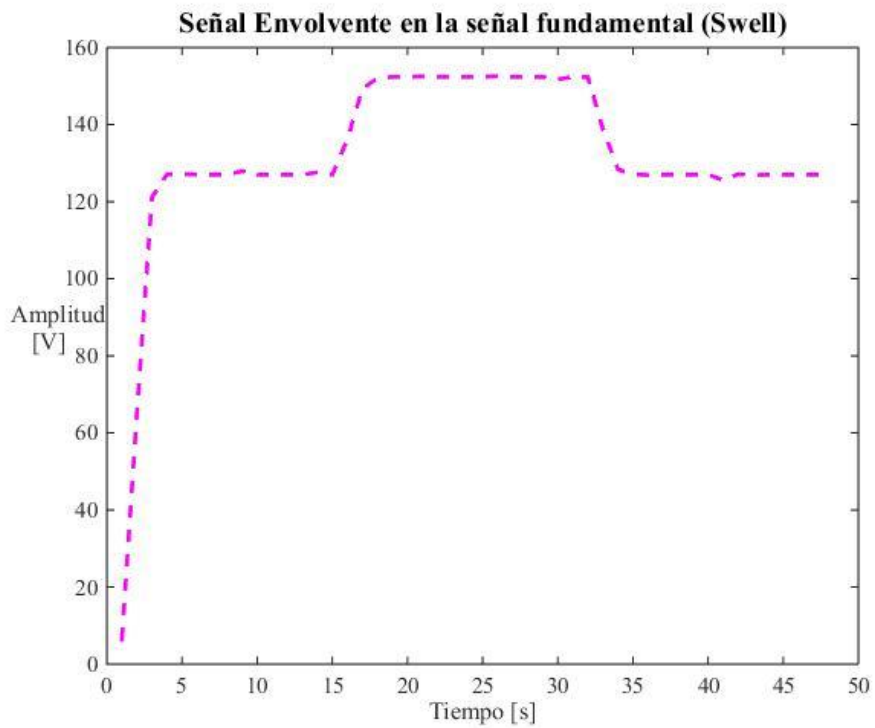
(b)

Figura 4.8. Análisis en la señal fundamental para señales de base de datos: (a) Sag y (b) Swell.

Por último el análisis muestra una señal envolvente con la amplitud de cada medio ciclo de la señal eléctrica, en esta señal envolvente se encuentra la información necesaria para determinar si ocurrió un disturbio eléctrico sag o swell al ver si existió un cambio en la amplitud de la señal, lo que en la envolvente se ve como un escalón. La Figura 4.9 y la Figura 4.10 muestran esta señal envolvente para los dos tipos de señales trabajadas.

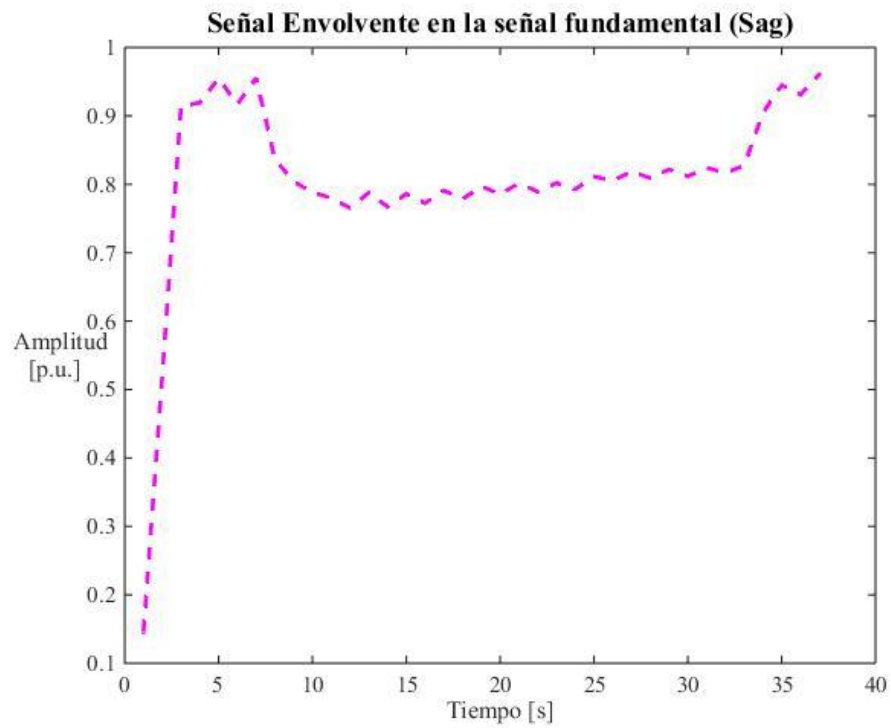


(a)

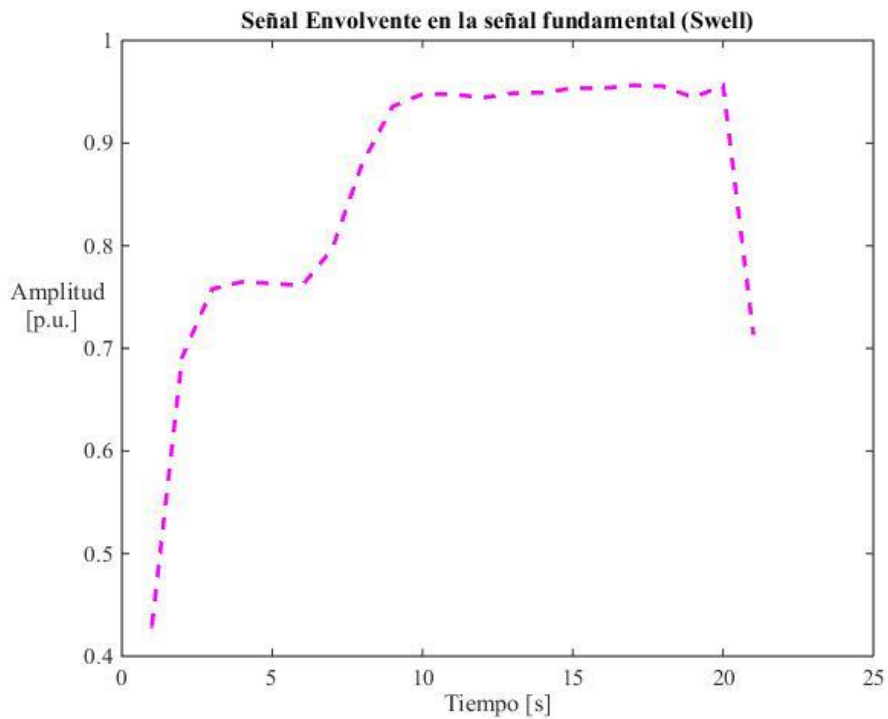


(b)

Figura 4.9. Señal Envolvente en la señal fundamental para señales sintéticas: (a) Sag y (b) Swell.



(a)



(b)

Figura 4.10. Señal Envolvente en la señal fundamental para señales de base de datos: (a) Sag y (b) Swell.

Esto se realizó para señales sintéticas que fueron generadas en un archivo script de MATLAB y para señales de base de datos tomadas de la IEEE y señales experimentales del grupo HSP Digital.

4.2 Resultados de la implementación en hardware

Una vez realizada la validación en software se trabajó sobre el sistema basado en FPGA para lo que se diseñaron las arquitecturas digitales en VHDL.

Se realizaron pruebas con las arquitecturas digitales obteniendo los siguientes resultados para distintas simulaciones.

En la Figura 4.11 se muestra la simulación para un disturbio eléctrico sag de amplitud 70% se puede ver como el MAG va determinando un mejor individuo para cada medio ciclo y en la parte del disturbio el mejor individuo es uno con una amplitud menor.

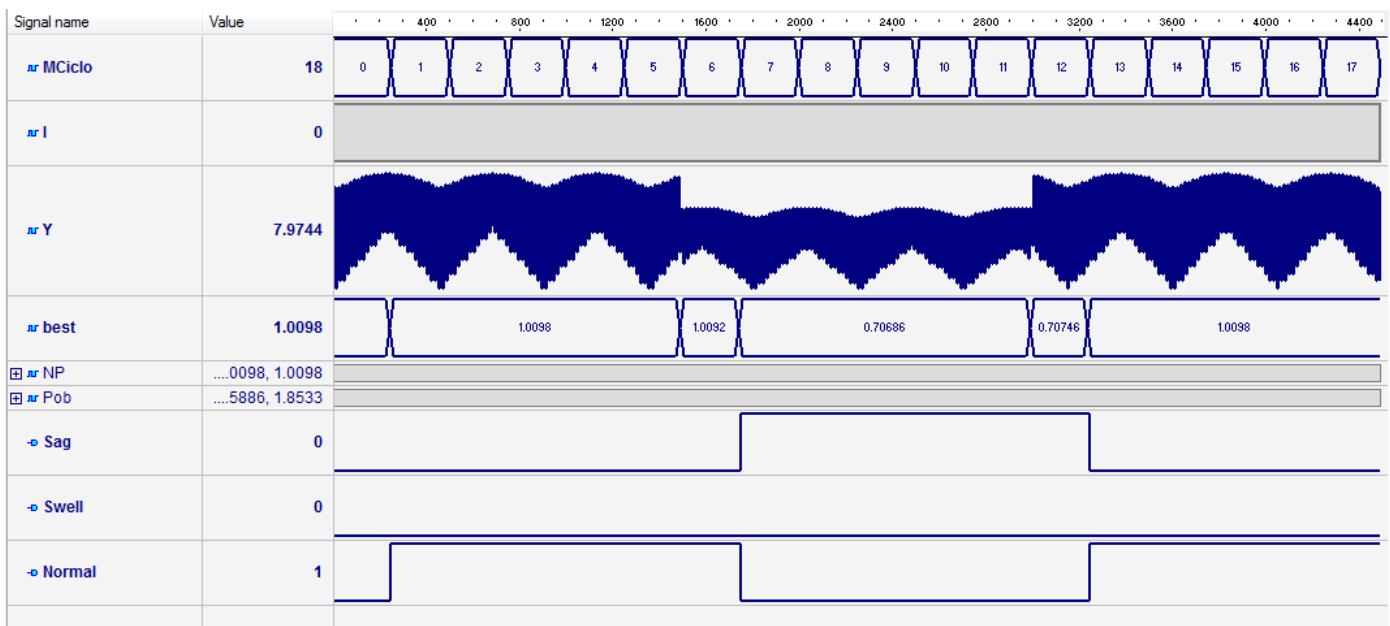


Figura 4.11. Simulación de las arquitecturas digitales para un disturbio sag.

La señal best, que representa el mejor individuo también puede verse como si fuera la señal envolvente que se generaba con los MAG en MATLAB, quedando de la forma en cómo se muestra en la Figura 4.12.

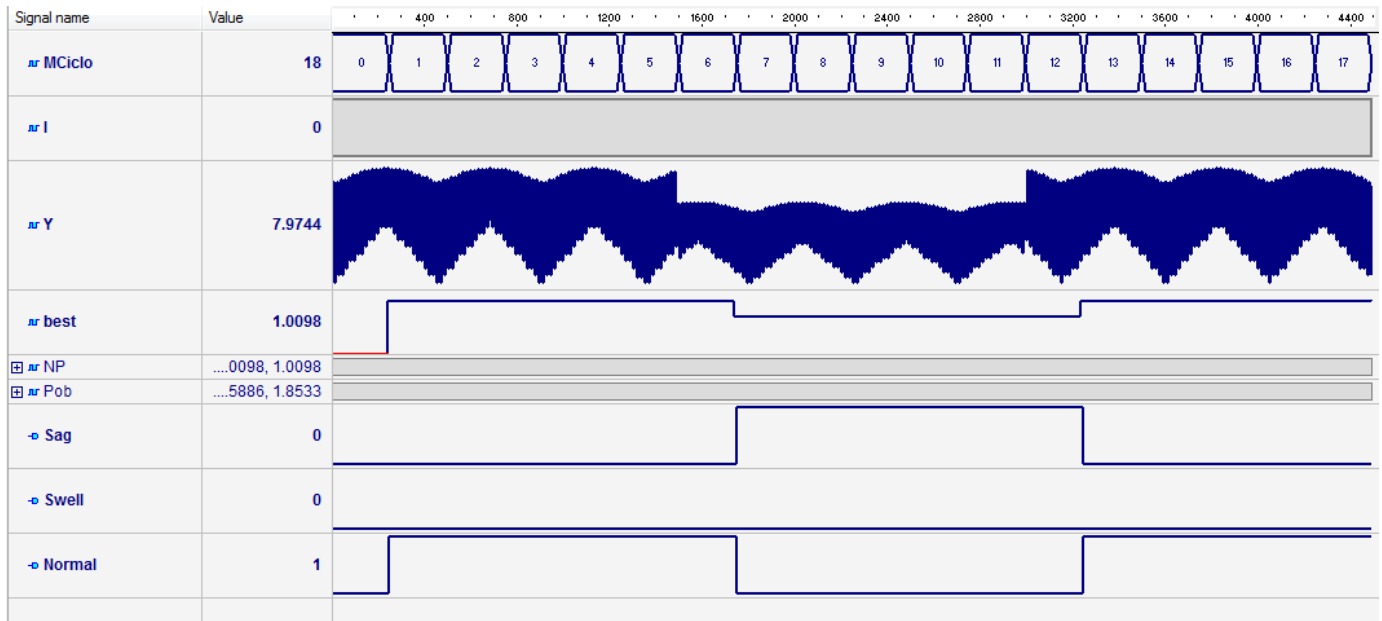


Figura 4.12. Señal envolvente en la simulación de las arquitecturas digitales para un disturbio sag.

De manera más detallada en la Figura 4.13, se muestra el proceso de los MAG en un medio ciclo en específico en este caso es el medio ciclo numero 1 donde el valor de la señal eléctrica no tenía ninguna afectación, se puede ver en esta figura como la población inicia aleatoriamente en un valor inicial para cada individuo y en cada iteración esta población va cambiando de tal manera que se encuentra el mejor individuo que cumple con la condición o función objetivo que es la del error absoluto vista en el capítulo 3, la Figura 4.14, es la continuación de las iteraciones en el medio ciclo número 1 donde se puede apreciar que en las últimas iteraciones el valor del mejor individuo ya converge en uno solo.

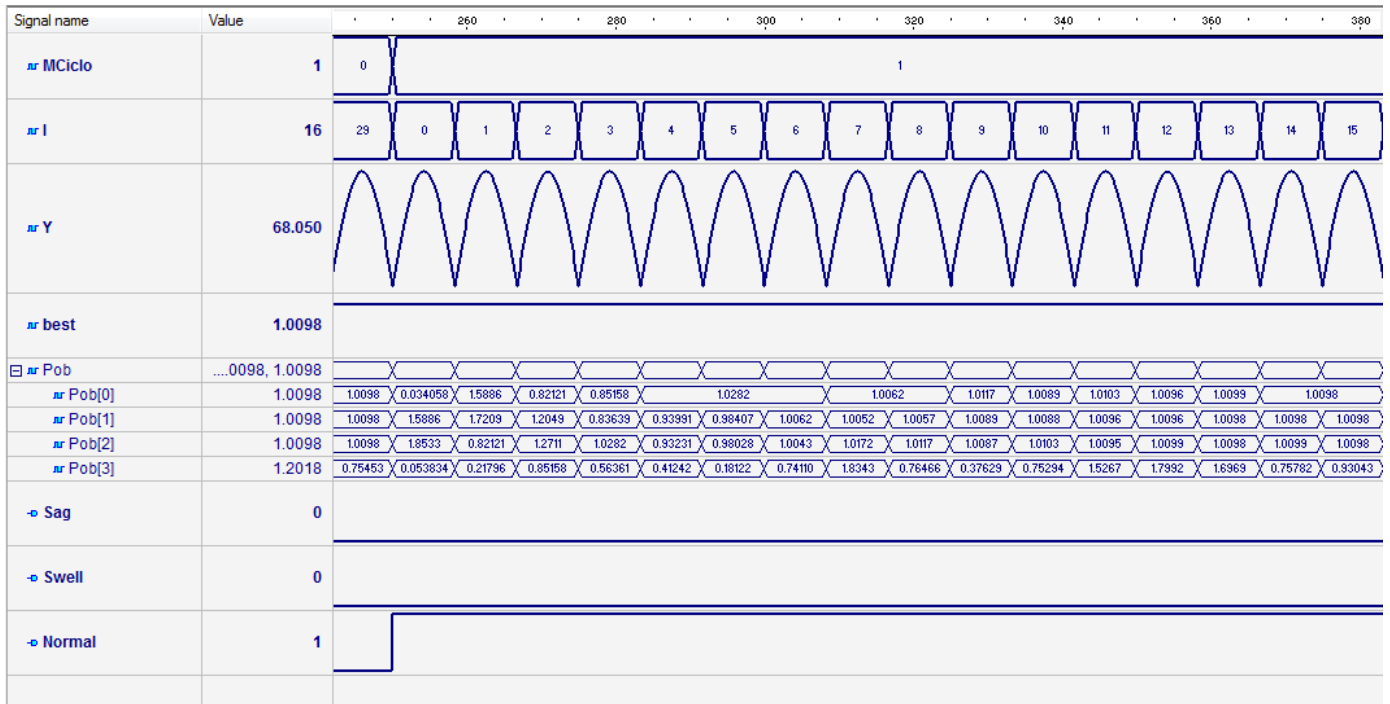


Figura 4.13. Evolución del método de MAG en la simulación de las arquitecturas digitales.

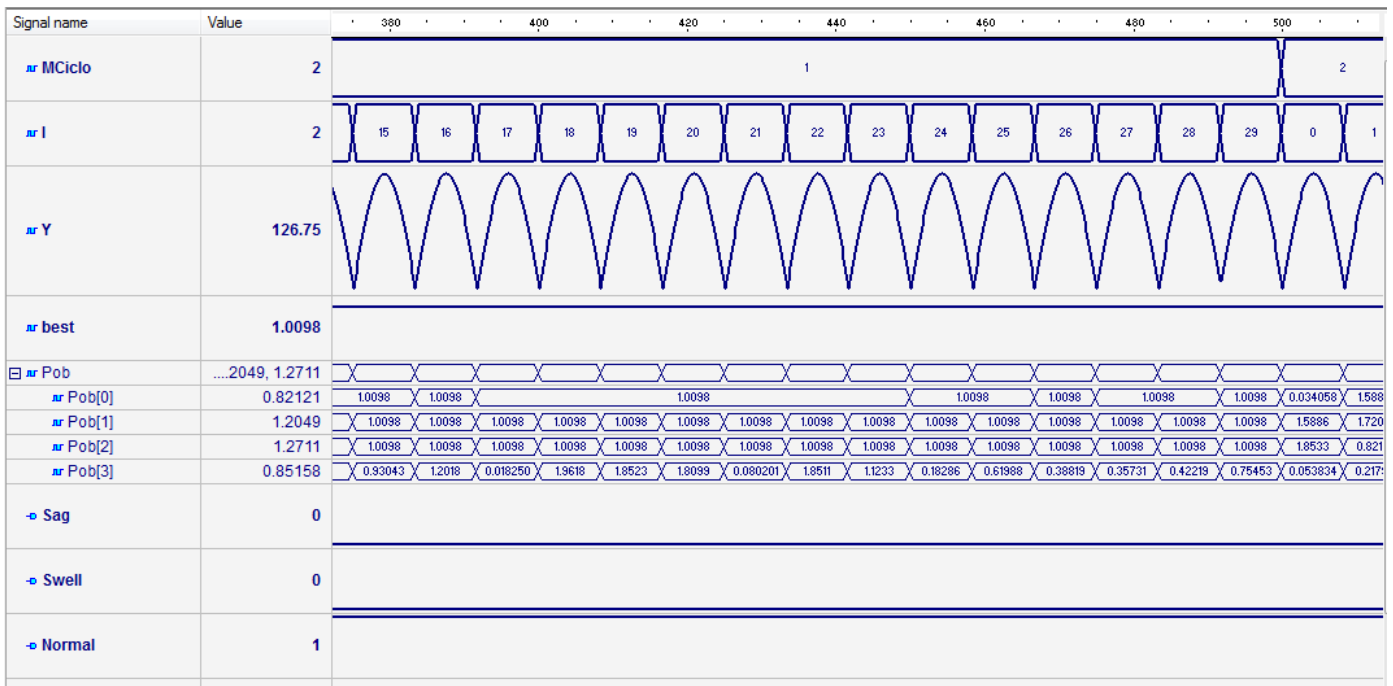


Figura 4.14. Convergencia del método de MAG en la simulación de las arquitecturas digitales.

En la Figura 4.15 se observa de manera más detallada el medio ciclo número 9 que es donde existió el disturbio eléctrico sag , se puede ver como ahora los individuos de la población convergen hacia el mejor valor que es diferente al del medio ciclo número 1, la Figura 4.16. Es la continuación de las iteraciones en este medio ciclo.

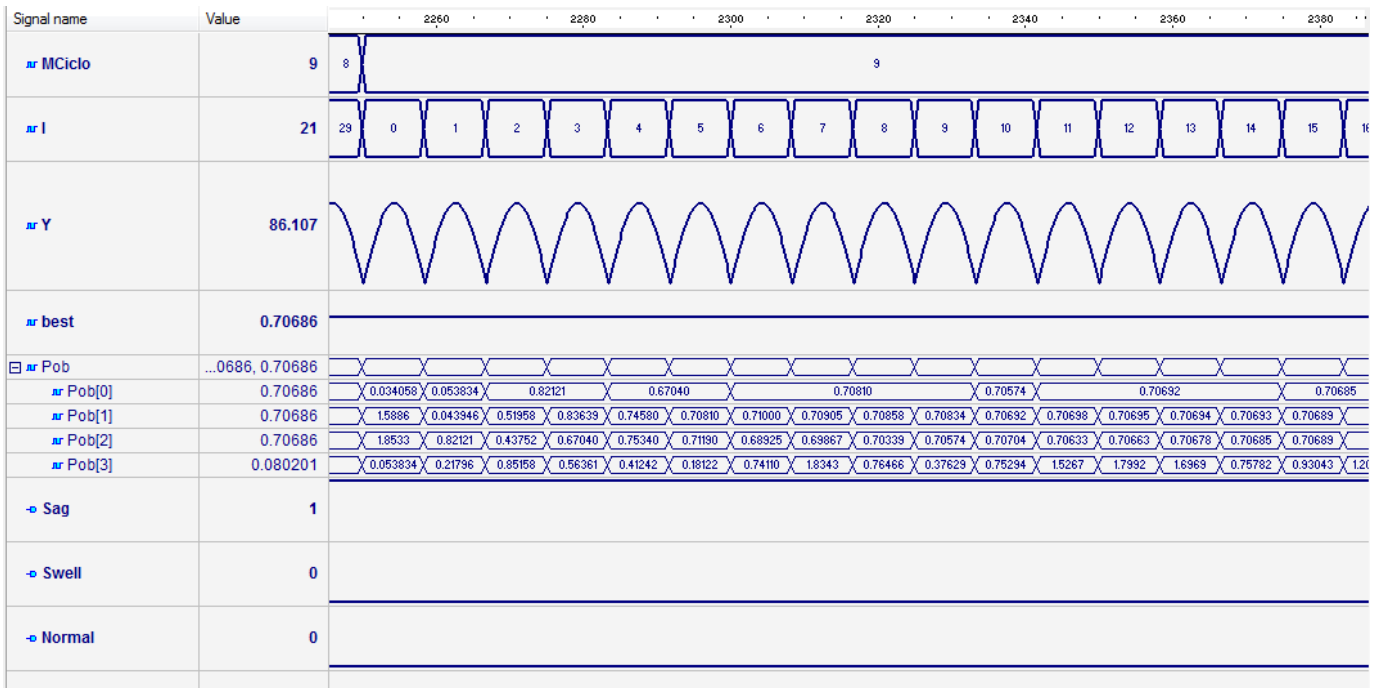


Figura 4.15. Evolución del método de MAG en la simulación de las arquitecturas digitales en un disturbio sag

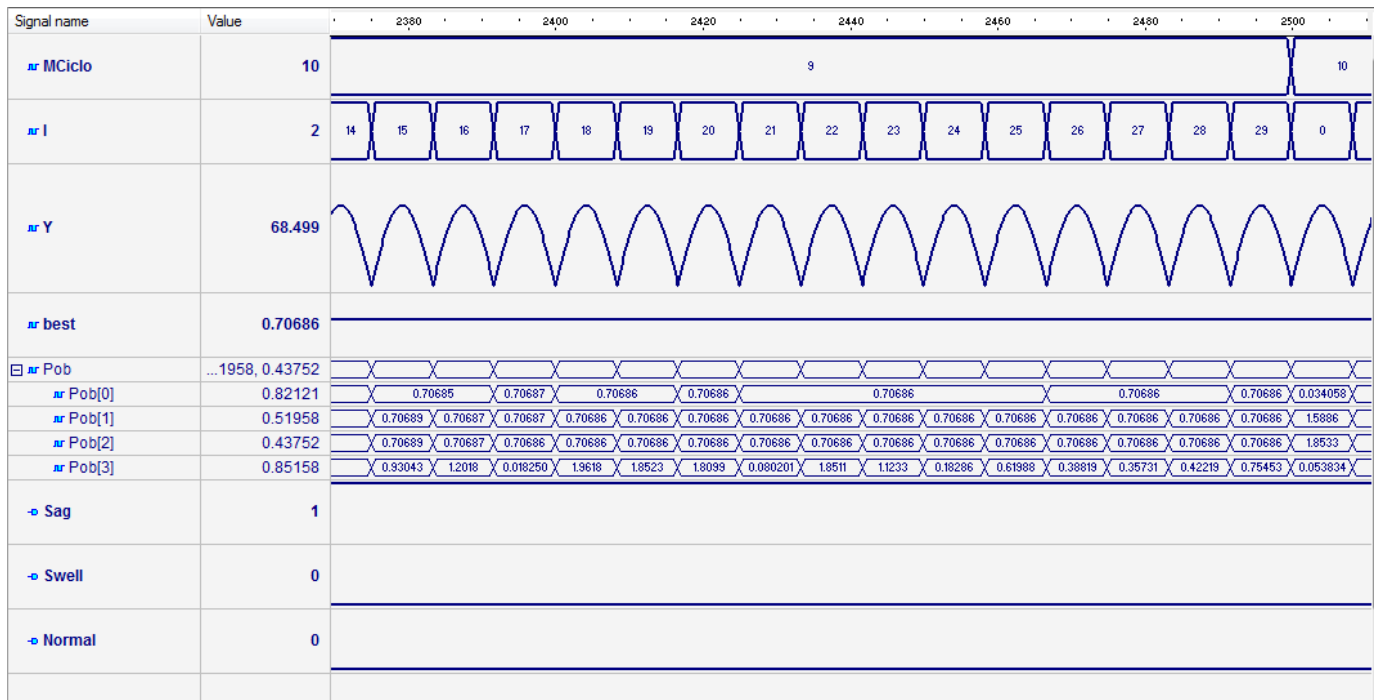


Figura 4.16. Convergencia del método de MAG en la simulación de las arquitecturas digitales en un disturbio sag

Para un disturbio swell se realizó la simulación con las arquitecturas digitales diseñadas, en la Figura 4.17 se muestra la señal y que es la que tiene el disturbio eléctrico swell de amplitud 30%, en la cual la señal best también indica el mejor individuo que existió en cada medio ciclo, y también se puede ver como la señal envolvente generada en MATLAB, esto se aprecia en la Figura 4.18.

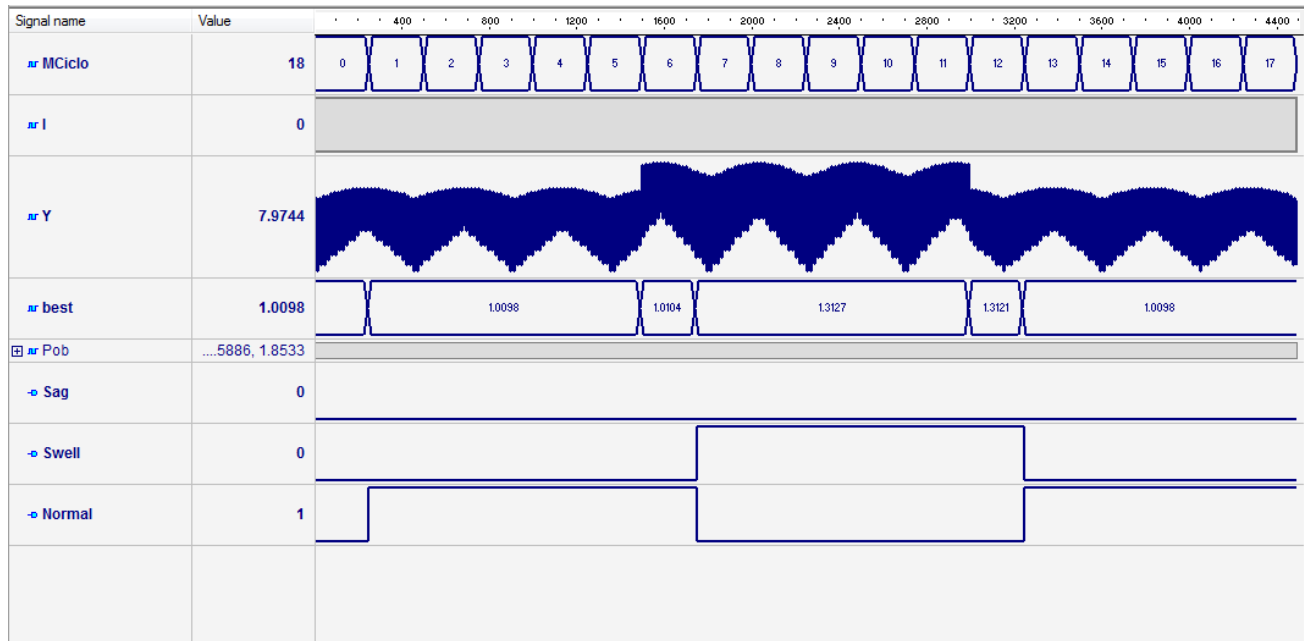


Figura 4.17. Simulación de las arquitecturas digitales para un disturbio swell.

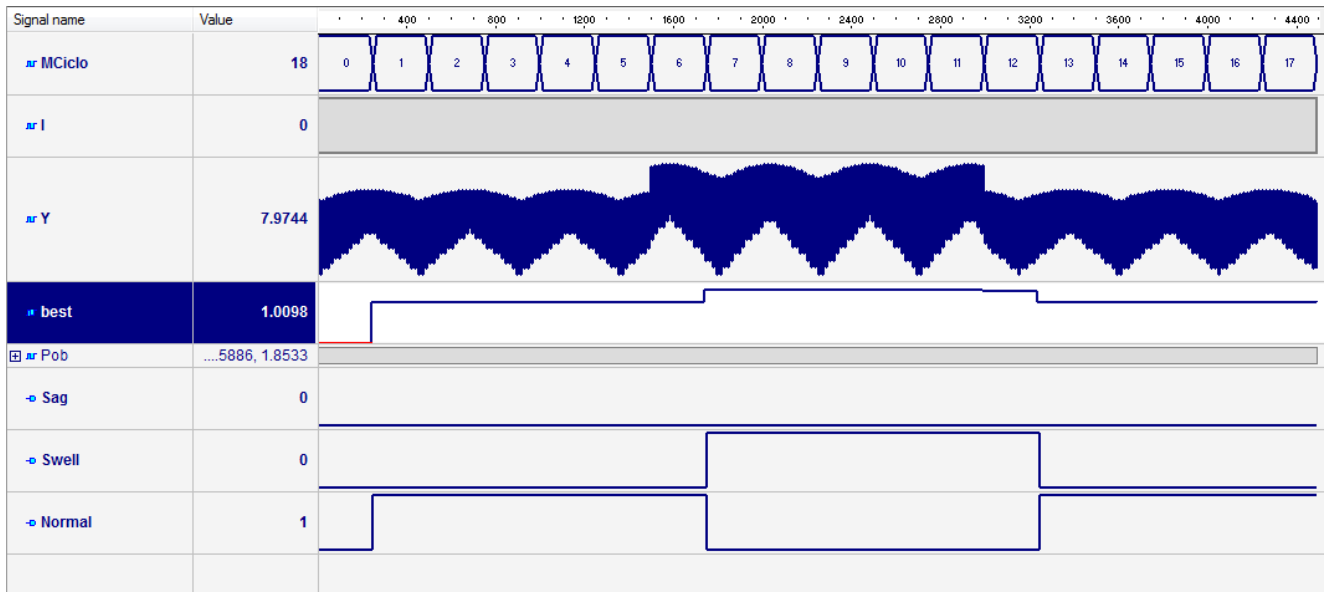


Figura 4.18. Señal envolvente en la simulación de las arquitecturas digitales para un disturbio swell

En la Figura 4.20. Se muestra a detalle el método de MAG en un medio ciclo que contiene el disturbio swell, se aprecia como la población va evolucionando cada medio ciclo hasta encontrar el mejor individuo que representa el menor error en la función objetivo, Figura 4.19, es la continuación de las iteraciones en ese medio ciclo donde se aprecia que el valor converge ya en las últimas iteraciones.

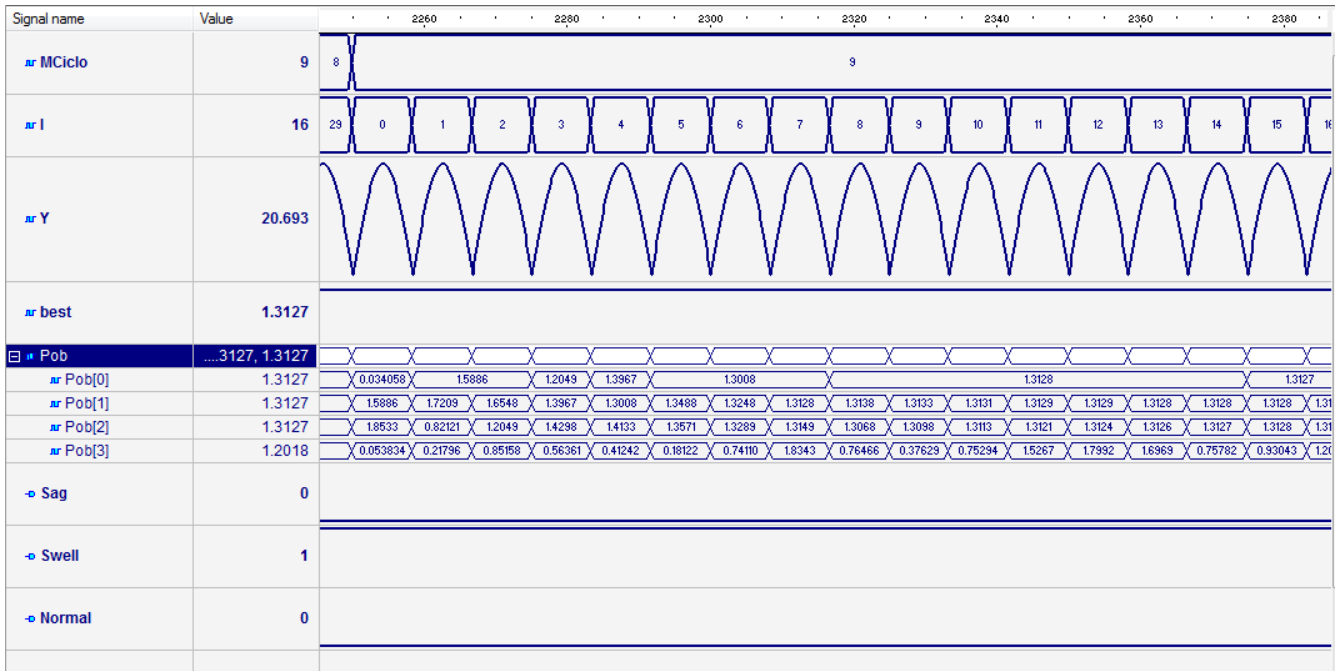


Figura 4.20. Evolución del método de MAG en la simulación de las arquitecturas digitales en un disturbio swell

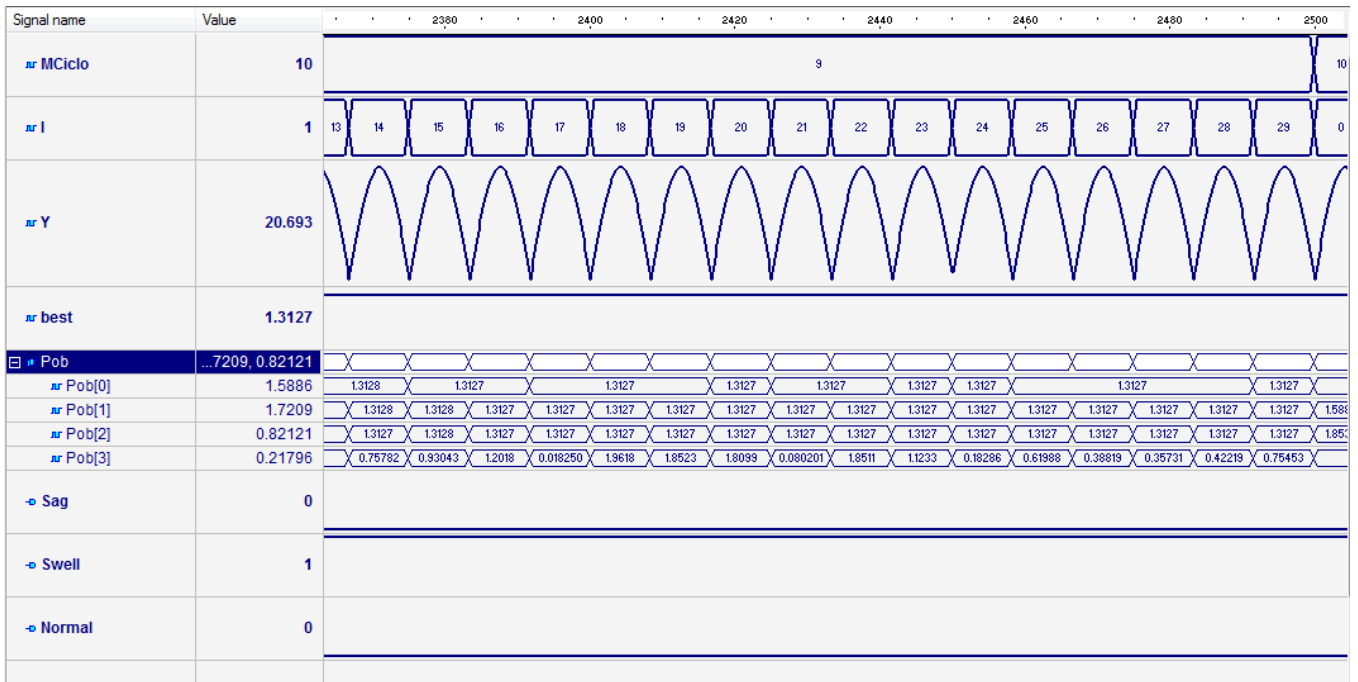


Figura 4.19. Convergencia del método de MAG en la simulación de las arquitecturas digitales en un disturbio swell

4.3 Resultados de pruebas sobre afectación de disturbios en máquina fresadora (Dinámica de Movimiento)

En el banco de pruebas se tenían dos cosas primero el sistema para la generación de disturbios en el cual se veía afectada solo una fase de la red eléctrica para generar un sag o swell.

La Figura 4.21 muestra la adquisición de las señales de voltaje sin ningún disturbio, en la Figura 4.22 se muestra la adquisición de las señales de voltaje cuando se le indujo un disturbio eléctrico sag en un intervalo de tiempo entre la adquisición. Para el disturbio swell la Figura 4.23 muestra la adquisición de las señales de voltaje cuando se generó este disturbio.

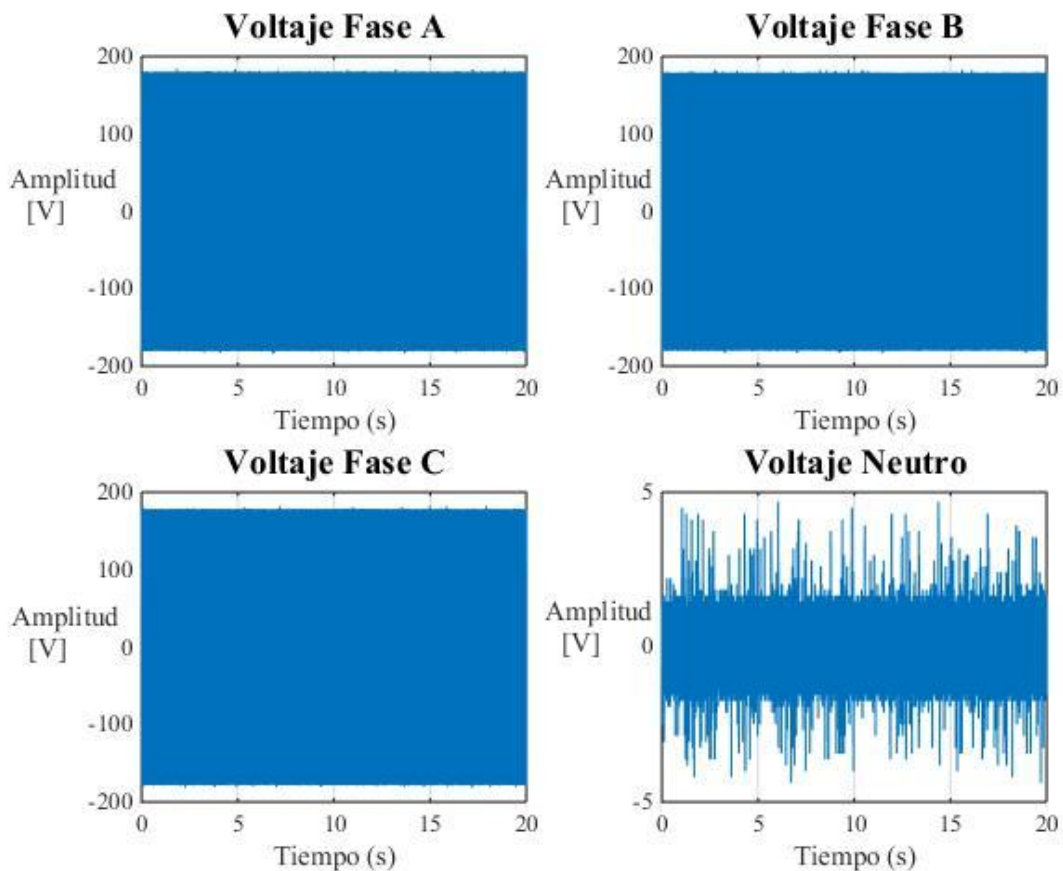


Figura 4.21. Adquisición de las señales de voltaje

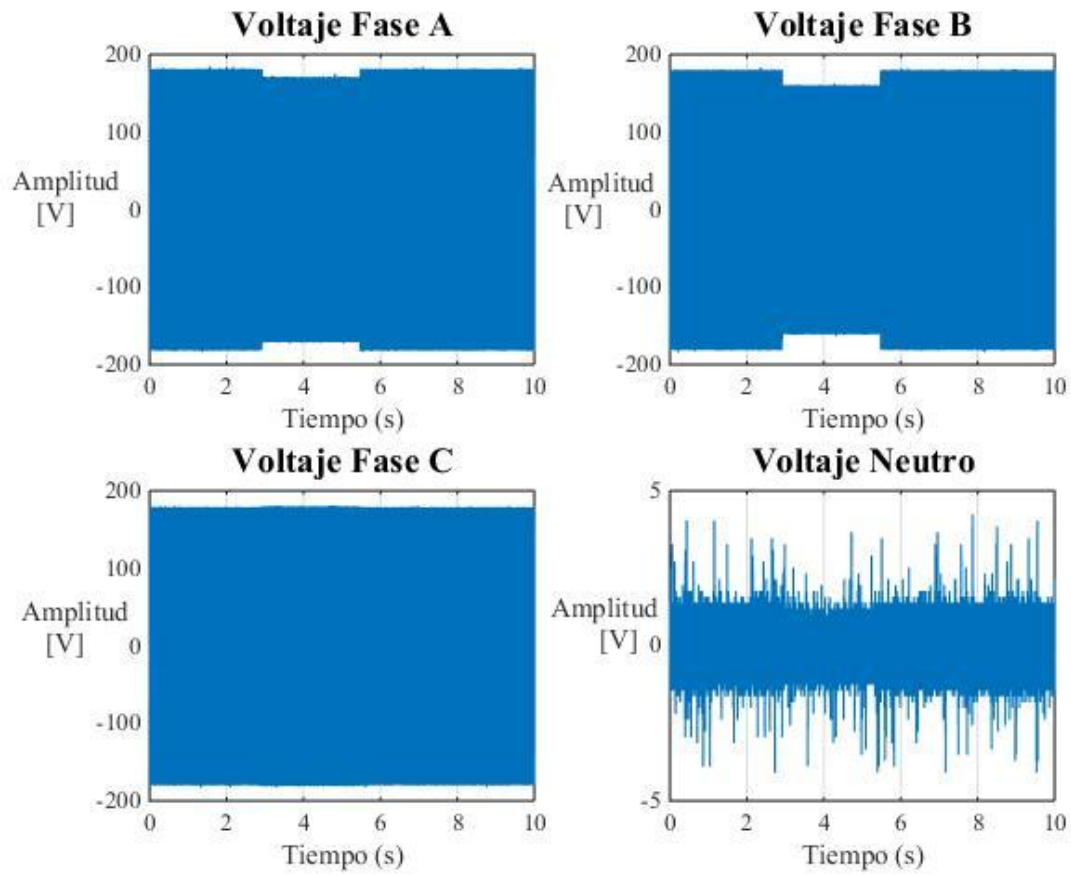


Figura 4.22. Adquisición de las señales de voltaje con un disturbio sag

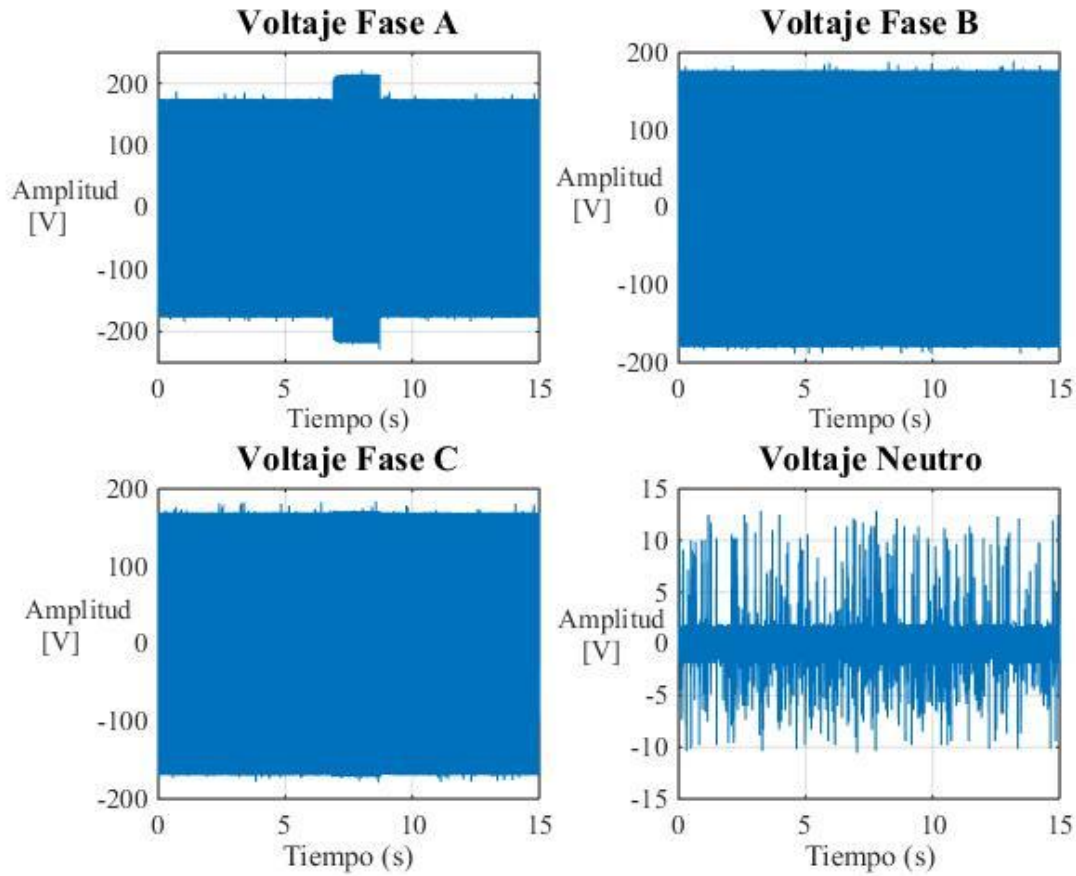


Figura 4.23. Adquisición de las señales de voltaje con un disturbio swell

La máquina fresadora realizaba un movimiento y cuando se movía se le inducía un disturbio después se obtenía la señal del encoder de la máquina fresadora, esta señal era solo la de posicionamiento así que se realizó un procesamiento para obtener los siguientes parámetros velocidad aceleración jerk, la Figura 4.24 muestra estos parámetros.

Como el disturbio eléctrico solo era de una duración aproximada entre 3 y 5 segundos solo se realizó el análisis en la zona donde se dio la afectación. La Figura 4.25 muestra el análisis de la referencia.

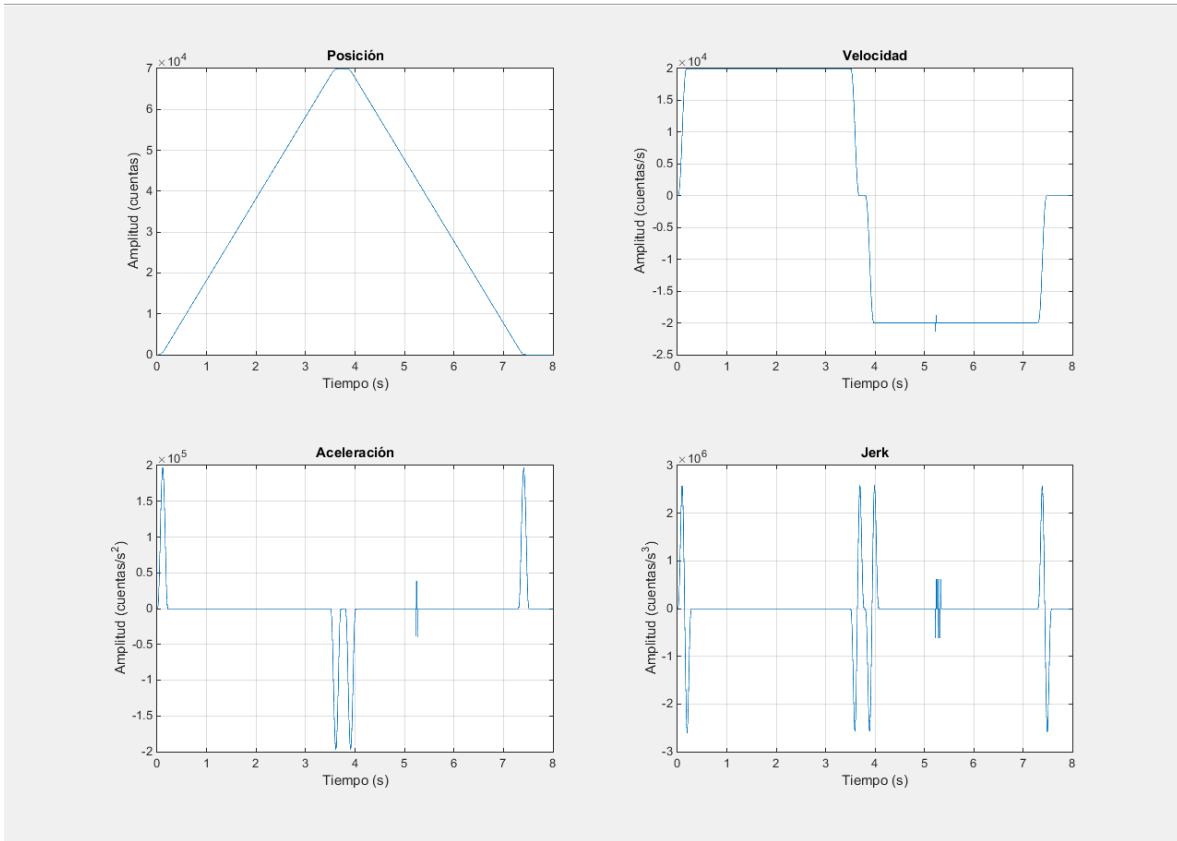


Figura 4.24. Señales de referencia de los parámetros de movimiento de la máquina fresadora

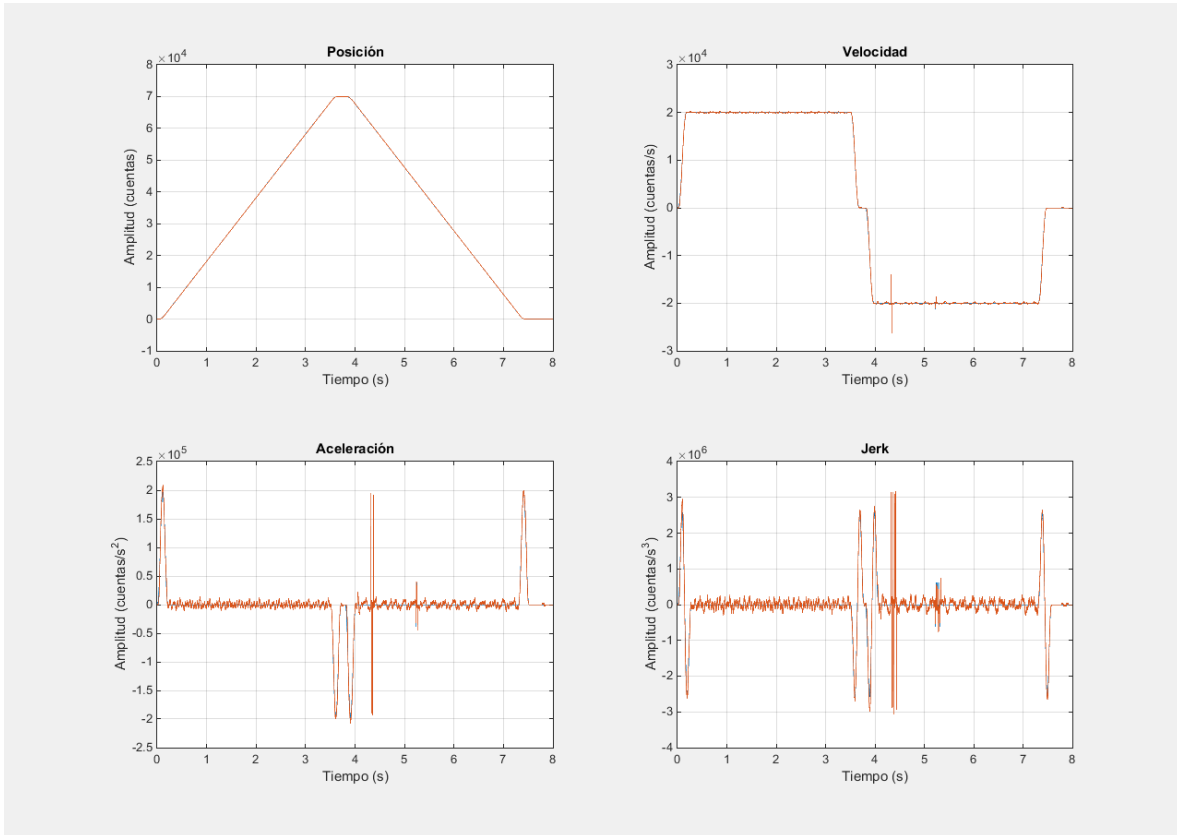


Figura 4.25. Señales de seguimiento de los parámetros de movimiento de la máquina fresadora

La Figura 4.26 muestra el análisis ante un disturbio sag de 90% de amplitud, la Figura 4.27 muestra el análisis ante un disturbio sag de 60% de amplitud, la Figura 4.28 muestra el análisis ante un disturbio sag de 75% de amplitud y la Figura 4.29 muestra el análisis ante un disturbio swell de 110% de amplitud.

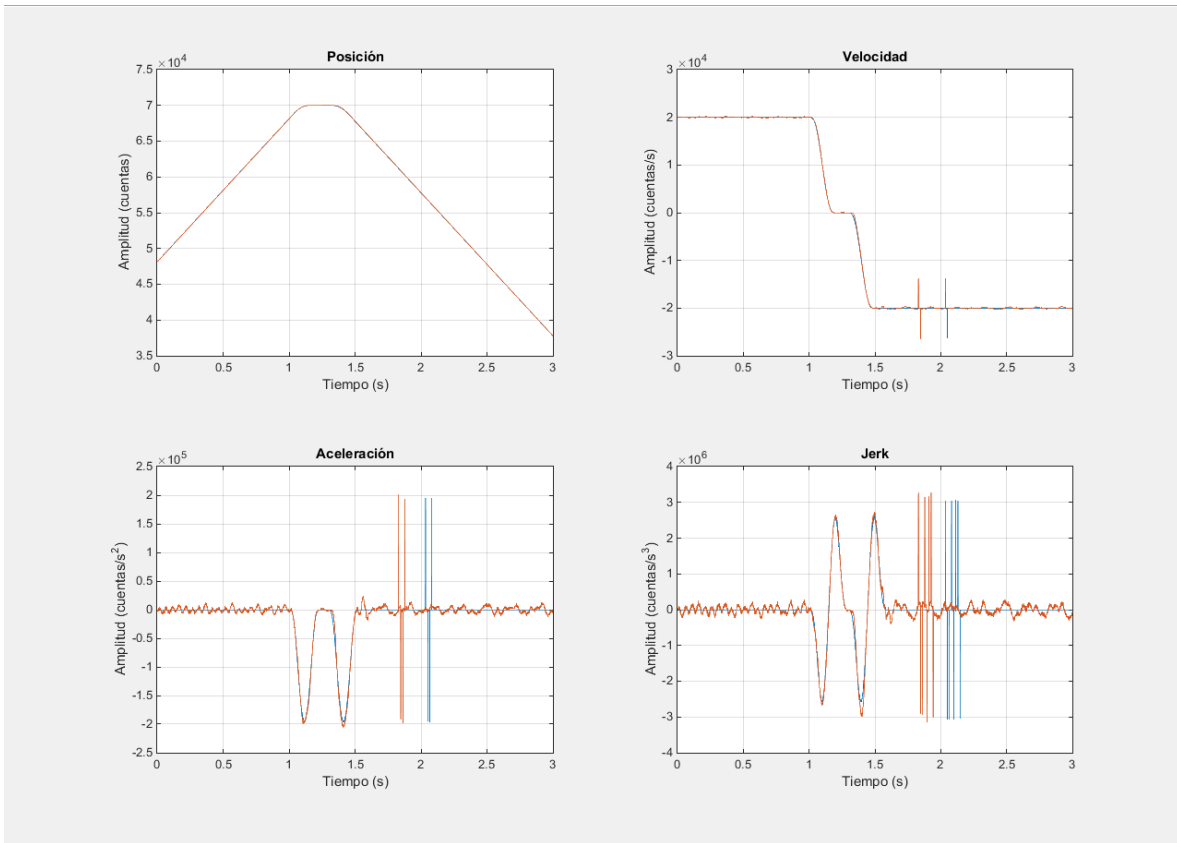


Figura 4.26. Señales de seguimiento de los parámetros de movimiento de la máquina fresadora en el momento de un disturbio Sag de 90% de amplitud

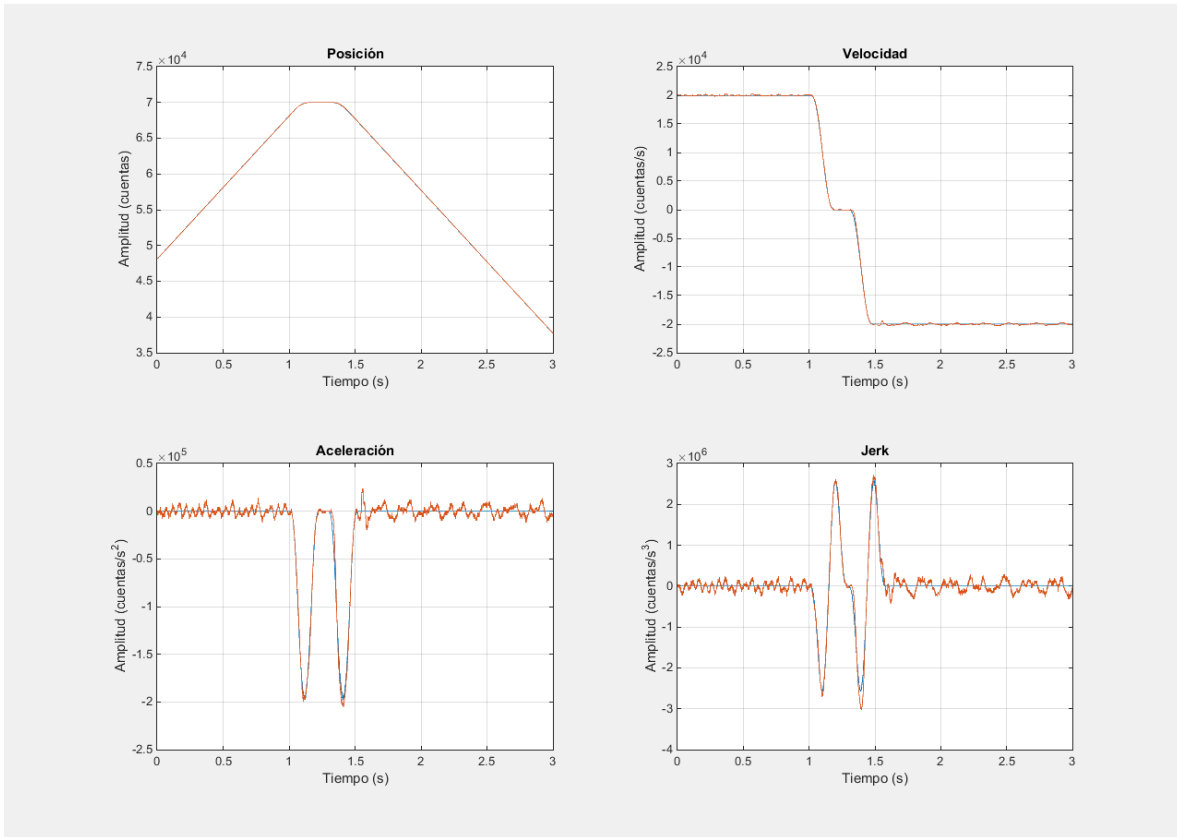


Figura 4.27. Señales de seguimiento de los parámetros de movimiento de la máquina fresadora en el momento de un disturbio Sag de 60% de amplitud

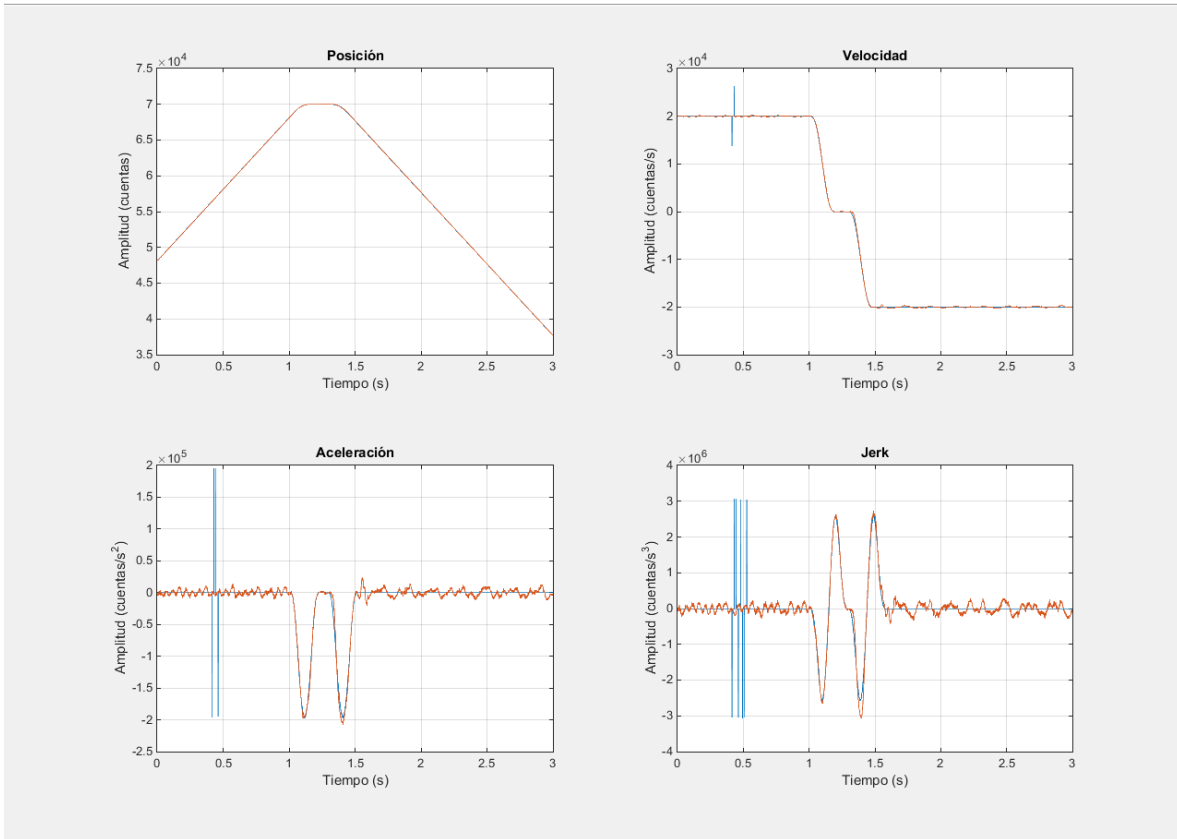


Figura 4.28. Señales de seguimiento de los parámetros de movimiento de la máquina fresadora en el momento de un disturbio Sag de 75% de amplitud

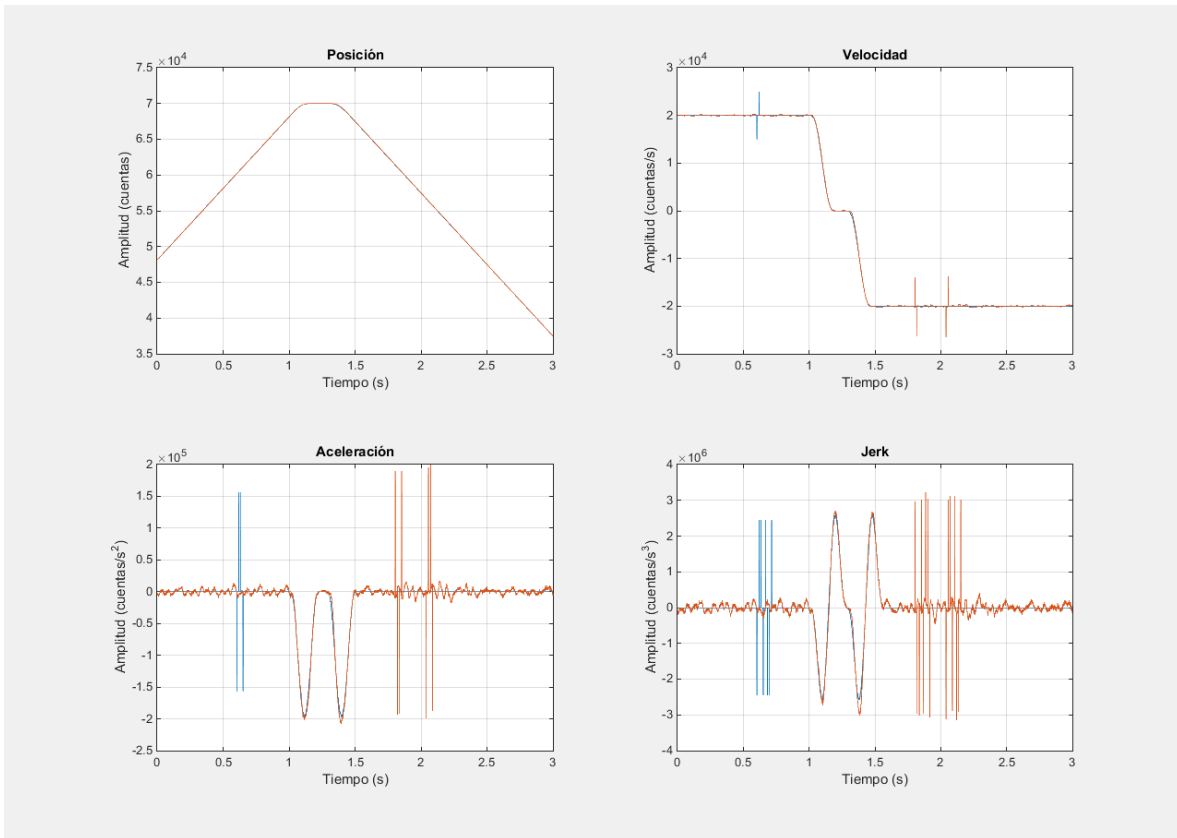


Figura 4.29. Señales de seguimiento de los parámetros de movimiento de la máquina fresadora en el momento de un disturbio Swell de 110% de amplitud

Con base a las pruebas realizadas anteriormente sin disturbio y con disturbio se estimó un error para cada parámetro, los cuales se muestran en las siguientes tablas.

Tabla 4.1. Valores estadísticos para las pruebas realizadas en el parámetro de posición

POSICIÓN			
Prueba	Desviación Estándar	IAE (integral del error absoluto)	ISE (integral del error cuadrático)
Referencia 1	21.9492	20.8179	493.7230
Referencia 2	21.5687	20.6086	476.4069
Referencia 3	22.1537	20.9062	503.6561
Referencia 4	21.9516	20.8274	492.7634
Referencia 5	22.0603	20.8529	498.3153
Prueba 1 (Sag 90%)	22.1426	20.7197	501.1506
Prueba 2 (Sag 90%)	21.8162	20.7560	487.3326
Prueba 3 (Sag 90%)	21.9696	20.7789	494.2070
Prueba 4 (Sag 90%)	21.7047	20.6354	482.5261
Prueba 5 (Sag 90%)	21.9711	20.7774	494.7237
Prueba 6 (Sag 60%)	22.0212	20.7767	496.9737
Prueba 7 (Sag 60%)	21.7352	20.6476	483.8947
Prueba 8 (Sag 60%)	21.7565	20.6741	485.2013
Prueba 9 (Sag 60%)	21.8673	20.7654	489.7892
Prueba 10 (Sag 60%)	21.8767	20.7200	490.0095
Prueba 11 (Sag 75%)	22.1735	20.9912	503.4721
Prueba 12 (Sag 75%)	21.3661	20.6377	468.5397
Prueba 13 (Sag 75%)	21.4605	20.6407	472.0988
Prueba 14 (Sag 75%)	22.4108	21.0573	514.3831
Prueba 15 (Sag 75%)	22.0756	21.1066	499.3486
Prueba 16 (Swell 110%)	21.7751	20.8439	485.5084
Prueba 17 (Swell 110%)	21.3742	20.8102	468.7285
Prueba 18 (Swell 110%)	21.8526	21.1605	490.0837
Prueba 19 (Swell 110%)	22.0072	20.9197	496.2275
Prueba 20 (Swell 110%)	21.4864	20.7167	473.0577

Tabla 4.2. Valores estadísticos para las pruebas realizadas en el parámetro de velocidad

Velocidad			
Prueba	Desviación Estándar	IAE (integral del error absoluto)	ISE (integral del error cuadrático)
Referencia 1	137.7240	89.9433	1.9001e4
Referencia 2	122.5107	87.7770	1.5046e4
Referencia 3	131.7121	93.0991	1.7388e4
Referencia 4	131.1427	92.8491	1.7232e4
Referencia 5	206.0081	96.5985	4.2471e4
Prueba 1 (Sag 90%)	366.3865	99.9833	1.3426e5
Prueba 2 (Sag 90%)	129.1498	92.6304	1.6715e4
Prueba 3 (Sag 90%)	236.3552	99.0877	5.5892e4
Prueba 4 (Sag 90%)	129.4780	93.6198	1.6799e4
Prueba 5 (Sag 90%)	235.8065	99.1085	5.5634e4
Prueba 6 (Sag 60%)	263.5510	100.8790	6.9486e4
Prueba 7 (Sag 60%)	129.7713	92.9741	1.6874e4
Prueba 8 (Sag 60%)	171.0644	94.4947	2.9297e4
Prueba 9 (Sag 60%)	132.1486	93.2136	1.7497e4
Prueba 10 (Sag 60%)	173.5254	95.6091	3.0141e4
Prueba 11 (Sag 75%)	177.2996	96.1402	3.1467e4
Prueba 12 (Sag 75%)	124.8700	89.6830	1.5635e4
Prueba 13 (Sag 75%)	168.7347	90.9536	2.8519e4
Prueba 14 (Sag 75%)	140.6431	91.6201	1.9816e4
Prueba 15 (Sag 75%)	136.4094	90.2975	1.8645e4
Prueba 16 (Swell 110%)	195.4555	88.4436	3.8248e4
Prueba 17 (Swell 110%)	119.3290	83.5277	1.4271e4
Prueba 18 (Swell 110%)	162.7604	85.4649	2.6535e4
Prueba 19 (Swell 110%)	406.2808	97.9420	1.6508e5
Prueba 20 (Swell 110%)	173.2485	96.6089	3.0052e4

Tabla 4.3. Valores estadísticos para las pruebas realizadas en el parámetro de aceleración

ACELERACIÓN			
Prueba	Desviación Estándar	IAE (integral del error absoluto)	ISE (integral del error cuadrático)
Referencia 1	5.2539e3	3.7012e3	2.7599e7
Referencia 2	4.6339e3	3.6088e3	2.1469e7
Referencia 3	4.8929e3	3.7526e3	2.3937e7
Referencia 4	4.9484e3	3.7845e3	2.4482e7
Referencia 5	8.5169e3	4.0094e3	7.2526e7
Prueba 1 (Sag 90%)	1.5913e4	4.1813e3	2.5318e8
Prueba 2 (Sag 90%)	4.8485e3	3.7403e3	2.3505e7
Prueba 3 (Sag 90%)	9.9918e3	4.1562e3	9.9819e7
Prueba 4 (Sag 90%)	4.8460e3	3.7764e3	2.3480e7
Prueba 5 (Sag 90%)	9.9379e3	4.1630e3	9.8745e7
Prueba 6 (Sag 60%)	1.1194e4	4.2717e3	1.2528e8
Prueba 7 (Sag 60%)	4.8686e3	3.7686e3	2.3702e7
Prueba 8 (Sag 60%)	6.9860e4	3.8744e3	4.8796e7
Prueba 9 (Sag 60%)	4.9585e3	3.7975e3	2.4583e7
Prueba 10 (Sag 60%)	7.0594e3	3.9492e3	4.9827e7
Prueba 11 (Sag 75%)	7.1828e3	3.9792e3	5.1584e7
Prueba 12 (Sag 75%)	4.7471e3	3.7100e3	2.2533e7
Prueba 13 (Sag 75%)	6.8557e3	3.8044e3	4.6994e7
Prueba 14 (Sag 75%)	5.2568e3	3.7406e3	2.7630e7
Prueba 15 (Sag 75%)	5.6405e3	3.7074e3	3.1811e7
Prueba 16 (Swell 110%)	8.2333e3	3.7360e3	6.7777e7
Prueba 17 (Swell 110%)	4.8590e3	3.4724e3	2.3606e7
Prueba 18 (Swell 110%)	6.7445e3	3.5964e3	4.5481e7
Prueba 19 (Swell 110%)	1.7749e4	4.2288e3	3.1499e8
Prueba 20 (Swell 110%)	7.7221e3	4.3333e3	5.9623e7

Tabla 4.4. Valores estadísticos para las pruebas realizadas en el parámetro de jerk

JERK			
Prueba	Desviación Estándar	IAE (integral del error absoluto)	ISE (integral del error cuadrático)
Referencia 1	1.2692e5	9.3948e4	1.6106e10
Referencia 2	1.1163e5	9.0353e4	1.2460e10
Referencia 3	1.1917e5	9.3531e4	1.4198e10
Referencia 4	1.1902e5	9.4528e4	1.4164e10
Referencia 5	1.9527e5	1.0173e5	3.8124e10
Prueba 1 (Sag 90%)	3.5420e5	1.0738e5	1.2544e11
Prueba 2 (Sag 90%)	1.1700e5	9.4238e4	1.3686e10
Prueba 3 (Sag 90%)	2.2689e5	1.0634e5	5.1472e10
Prueba 4 (Sag 90%)	1.1712e5	9.4228e4	1.3714e10
Prueba 5 (Sag 90%)	2.2574e5	1.0637e5	5.0949e10
Prueba 6 (Sag 60%)	2.5419e5	1.1057e5	6.4604e10
Prueba 7 (Sag 60%)	1.1753e5	9.4589e4	1.3811e10
Prueba 8 (Sag 60%)	1.6124e5	9.7599e4	2.5994e10
Prueba 9 (Sag 60%)	1.1976e5	9.5336e4	1.4341e10
Prueba 10 (Sag 60%)	1.6215e5	9.9501e4	2.6288e10
Prueba 11 (Sag 75%)	1.6674e5	1.0074e5	2.7799e10
Prueba 12 (Sag 75%)	1.1265e5	9.1426e4	1.2688e10
Prueba 13 (Sag 75%)	1.5850e5	9.6028e4	2.5119e10
Prueba 14 (Sag 75%)	1.2548e5	9.3500e4	1.5741e10
Prueba 15 (Sag 75%)	1.2839e5	9.1024e4	1.6481e10
Prueba 16 (Swell 110%)	1.8895e5	9.6953e4	3.5697e10
Prueba 17 (Swell 110%)	1.1137e5	8.5674e4	1.2401e10
Prueba 18 (Swell 110%)	1.5416e5	8.8669e4	2.3761e10
Prueba 19 (Swell 110%)	3.9533e5	1.1020e5	1.5626e11
Prueba 20 (Swell 110%)	1.5748e5	9.0068e4	2.4796e10

Conclusiones y perspectivas

El método de MAG resultó eficiente para la detección de los dos tipos de disturbios eléctricos a los cuales se enfocó este trabajo, los resultados tanto en software para validación como la simulación de las arquitecturas digitales diseñadas para una plataforma en FPGA fueron exitosos.

El sistema de generación de disturbios eléctricos sag y swell permitió realizar distintas pruebas para analizar la afectación que se daba con esos disturbios a una máquina fresadora, a pesar de las limitaciones físicas del sistema que impedían suministrar valores muy altos o bajos en una fase, el rango que se pudo manejar fue suficiente para realizar distintas pruebas.

Las pruebas realizadas con el disturbio sag, presentaban que si había una afectación en los parámetros de movimiento de la máquina, que a pesar que en algunos podían ser mínimos, estos podían representar como fallas en el acabado de una pieza así como inexactitud en el movimiento de la máquina y en operaciones que requieran de mucha exactitud podían ser muy significativos.

En las pruebas realizadas con el disturbio swell se determinó que a partir de un rango superior al 110% de la amplitud del voltaje nominal, la afectación que generaba este disturbio era de enclavar la máquina, porque no terminaba de realizar el movimiento que se estaba realizando y se bloqueaba la comunicación con los elementos de la máquina, servomotores y la interfaz que permitía la comunicación con la máquina.

El análisis de la afectación de estos disturbios se realizó mediante procesamiento de señales, el cual es un tema de interés en la actualidad por todo lo que se puede realizar y determinar mediante esta técnica.

El sistema para la detección de disturbios eléctricos basado en FPGA es de arquitectura abierta para poder realizar futuras mejoras y versiones posteriores que puedan tener como base el sistema aquí diseñado.

Referencias

- de Santiago-Pérez, J. J., Osornio-Rios, R. A., Romero-Troncoso, R. J., Herrera-Ruiz, G. , Delgado-Rosas, M. (2008). DSP algorithm for the extraction of dynamics parameters in CNC machine tool servomechanisms from an optical incremental encoder, *International Journal of Machine Tools and Manufacture*, Volume 48, Issues 12–13, Pages 1318-1334.
- García Ramírez, Armando Guadalupe 2014. Sensores colaborativos para detección de fallas en máquinas eléctricas. Universidad Autónoma de Querétaro.
- Granados Lieberman, David 2013. Análisis en maquinaria CNC ante variaciones de bajo voltaje y sus efectos en la calidad de la energía. Universidad Autónoma de Querétaro.
- Granados-Lieberman, D., Valtierra-Rodriguez, M., Morales-Hernández, L. A., Romero-Troncoso, R. J., & Osornio-Rios, R. A. (2013). A Hilbert Transform-Based Smart Sensor for Detection, Classification, and Quantification of Power Quality Disturbances. *Sensors (14248220)*, 13(5), 5507-5527.
- Gutiérrez Torres, Claudia 2014. Sistema de generación y monitoreo de disturbios eléctricos. Universidad Autónoma de Querétaro.
- IEEE Std 1159-2009. IEEE Recommended Practice for Monitoring Electric Power Quality.
- Jaen Cuellar, Arturo Yosimar 2011. Desarrollo de Perfiles Polinomiales 3D basado en FPGA para Control de Posición en Máquina Fresadora CNC. Universidad Autónoma de Querétaro.
- Jaen Cuellar, Arturo Yosimar 2014. Diseño de arquitecturas basadas en FPGA para algoritmos genéticos, aplicado a la optimización de controladores en sistemas de posicionamiento. Universidad Autónoma de Querétaro.
- Luna Ortiz, Carlos Ricardo 2011. Diseño e implementación de un algoritmo genético en FPGA para sintonización de controladores PID. Universidad Autónoma de Querétaro.

- Morales-Velazquez, L., Romero-Troncoso, R. J., Osornio-Rios, R. A., Herrera-Ruiz, G., Cabal-Yepez, E. (2010) Open-architecture system based on a reconfigurable hardware–software multi-agent platform for CNC machines, *Journal of Systems Architecture*, Volume 56, Issue 9, Pages 407-418.
- Osornio-Rios, R. A., Romero-Troncoso, R. J., Herrera-Ruiz, G., Castañeda-Miranda, R. (2007). Computationally efficient parametric analysis of discrete-time polynomial based acceleration–deceleration profile generation for industrial robotics and CNC machinery, *Mechatronics*, Volume 17, Issue 9, Pages 511-523.
- Osornio-Rios, R. A., Romero-Troncoso, R. J., Herrera-Ruiz, G. y Castañeda-Miranda, R. (2008). “The application of reconfigurable logic to high speed CNC milling machine controllers,” *Control Engineering Practice*, vol. 16, no. 6, pags: 674 – 684.
- Pérez Cruz, Ángel 2009. Diseño óptimo de una microviga para mejorar su sensibilidad por medio de algoritmos genéticos. Universidad Autónoma de Querétaro.
- Rivera Guillen, Jesus Rooney 2010. Algoritmos Óptimos en FPGA para Controlar la Dinámica de movimiento en maquinaria CNC. Universidad Autónoma de Querétaro.
- Tomáš Radil, Fernando M. Janeiro, Pedro M. Ramos, A. Cruz Serra, (2008) "An efficient approach to detect and classify power quality disturbances", *COMPEL - The international journal for computation and mathematics in electrical and electronic engineering*, Vol. 27 Iss: 5, pp.1178 – 1191.

Anexos

Anexo 1. Listado de la arquitectura general para el sistema basado en FPGA para detección de Sag y Swell mediante MAG

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3
4  package Poblacion is
5      constant n: integer :=3;
6      constant t: integer :=29;
7      type P is array (0 to n) of real;
8      type R is array (0 to t) of Real;
9  end package Poblacion;
10
11 library ieee;
12 use ieee.std_logic_1164.all;
13 use ieee.numeric_std.all;
14 use IEEE.MATH_REAL.ALL;
15 use Poblacion.all;
16 library std;
17 use std.textio.all;
18
19 entity SBF is port(
20     clk: in std_logic;
21     rst: in std_logic;
22     z: in real;
23     NumM: in integer;
24     Sag: out std_logic;
25     Swell: out std_logic;
26     Normal: out std_logic
27 );
28 end SBF;
29
30 architecture Behavior of SBF is
31
32 component PobIni is port(
33     clk: in std_logic;
34     rst: in std_logic;
35     PobI: out P
36 );
37 end component;
38
```

```

39 component Cordic is port(
40     clk: in std_logic;
41     rst: in std_logic;
42     I: in integer;
43     z: in real;
44     PobI: in P;
45     NP: in P;
46     sin: out real;
47     PobEva: out P);
48 end component;
49
50 component LUT_Seno is port(
51     clk: in std_logic;
52     rst: in std_logic;
53     X: in integer;
54     Y: out real
55 );
56 end component;
57
58 component Gener_Error is port
59 (
60     clk: in std_logic;
61     rst: in std_logic;
62     PobEva: in P;
63     Y: in real;
64     H: in integer;
65     Error: out P;
66     ErrorAbs: out P;
67     SumErrorAbs: out P;
68     ENABLEordena: out std_logic;
69     ENABLEope: out std_logic
70 );
71 end component;
72
73
74 component Ordena is port(
75     clk: in std_logic;
76     rst: in std_logic;

```

```

77     Enable: in std_logic;
78     SumErrorAbs: in P ;
79     Pob: in P;
80     SumErrorAbsOrd: out P;
81     PobOrd: OUT P
82     );
83 end component;
84
85 component OperGenet is port
86     (
87     clk: in std_logic;
88     rst: in std_logic;
89     Enable: in std_logic;
90     I: in integer;
91     Pob: in P;
92     NP: out P
93     );
94 end Component;
95
96
97
98 COMPONENT Contador_de_Iter is
99 port(
100     clk : in std_logic;
101     rst : in std_logic;
102     NumM: IN INTEGER;
103     X: out integer;
104     I :OUT INTEGER;
105     MCiclo: out integer
106     );
107 end COMPONENT;
108
109 component Ind_Disturbio is
110 port(
111     clk : in std_logic;
112     rst : in std_logic;
113     Mciclo: in integer;
114     best: in real;

```

```

115     normal: out std_logic;
116     sag : out std_logic;
117     swell : out std_logic
118 );
119 end component;
120
121 signal MCiclo,I: integer;
122 signal X: integer;
123 signal Y,sin,best: real;
124 signal PobEva,PobI,PobOrd,NP,Pob: P;
125 signal EnableOrdena,EnableOpe: std_logic;
126 signal Error,ErrorAbs,SumErrorAbs,SumErrorAbsOrd: P;
127
128
129 begin
130
131     B0: PobIni port map (clk,rst,PobI);
132     B1: Cordic port map (clk,rst,I,z,PobI,NP,sin,PobEva);
133     B2: LUT_Seno port map (clk,rst,X,Y);
134
135     B3: Gener_Error port map (clk,rst,PobEva,Y,NumM,Error,ErrorAbs,SumErrorAbs,ENABLEordena,eENABLEope);
136
137     B4: Ordena port map (clk,rst,EnableOrdena,SumErrorAbs,Pob,SumErrorAbsOrd,PobOrd);
138
139     B5: OperGenet port map (clk,rst,EnableOpe,I,PobOrd,NP);
140
141     B6: Contador_de_Iter port map (clk,rst,NumM,X,I,MCiclo);
142
143     B7: Ind_Disturbio port map (clk,rst,Mciclo,best,normal,sag,swell);
144
145     process (I)
146     begin
147
148         if I=0 THEN
149             Pob<=PobI;
150         else
151             Pob<=NP;
152         end if;
153
154         if I=29 then
155             BEST<=NP(0);
156         end if;
157
158     end process;
159
160 end behavior;

```