



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

Procesamiento de Imágenes de Mastografía con Implementación en Hardware

TESIS

**Para optar por el grado de:
Licenciado en Ciencias de la Computación**

PRESENTA

Victor Hugo Arroyo Domínguez

AGOSTO DE 2008



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

**Procesamiento de Imágenes de Mastografía
con Implementación en Hardware**

TESIS

**Para optar por el grado de:
Licenciado en Ciencias de la Computación**

PRESENTA

Victor Hugo Arroyo Domínguez

ASESOR

Dr. José Guadalupe Vázquez Luna

COASESOR

Dr. Manuel Martín Ortiz

AGOSTO DE 2008

Éste trabajo fue realizado con apoyo de CONACyT
mediante el proyecto SALUD-2004-01-021

*A mis **padres**,
por su apoyo moral y económico.*

*A **Antonia**,
por su apoyo y solidaridad incondicional.*

AGRADECIMIENTOS

Un agradecimiento a mi **familia** y **amigos** quienes siempre han estado en los momentos alegres y en los momentos difíciles.

Un agradecimiento al **Dr. José Guadalupe Vázquez Luna** por su conocimiento, experiencia, dedicación y apoyo en la dirección de este trabajo de tesis. Al **Dr. Manuel Martín Ortiz** por el apoyo brindado para la realización de este trabajo de tesis.

Un agradecimiento especial a **CONACyT** por el apoyo otorgado durante la realización de este trabajo.

RESUMEN

El desarrollo tecnológico ha propiciado un cambio asombroso en la medicina, el uso de nueva tecnología en el diagnóstico y tratamiento de diversas enfermedades ha significado la optimización del proceso de atención en el paciente, dando como resultado una mayor esperanza de vida.

Adentrándonos en el área de oncología y con la intención de facilitar el diagnóstico del cáncer de mama, se pretende desarrollar un conjunto de herramientas con las cuales se agilicen y perfeccionen las técnicas de diagnóstico de dicha enfermedad.

Para el diagnóstico del cáncer de mama, el método más extendido es la mastografía en el cual se utilizan rayos X para obtener imágenes de la glándula mamaria, en dicha imagen es posible localizar alteraciones morfológicas que le permiten, al radiólogo, inferir sobre la presencia o ausencia de diversas patologías. Existen sin embargo un elevado número de casos donde es difícil realizar un diagnóstico acertado debido a lo difuso de las imágenes, por lo cual el trabajo en el área del procesamiento digital para aplicaciones en la detección temprana del cáncer de mama es una necesidad imperiosa.

Este desarrollo forma parte integral del proyecto “*Diseño y construcción de un mastógrafo digital*” con número de referencia *CONACYT SALUD-2004-01-021*.

TABLA DE CONTENIDO

AGRADECIMIENTOS	I
RESUMEN	II
TABLA DE CONTENIDO	III
ÍNDICE DE FIGURAS	V
I. INTRODUCCIÓN	1
I.1 INTRODUCCIÓN.....	1
I.2 OBJETIVOS.....	1
I.3 RESULTADOS ESPERADOS	1
I.4 ORGANIZACIÓN DE LA TESIS	2
II. MARCO TEÓRICO.....	3
II.1 INTRODUCCIÓN.	3
II.2 CANCER DE MAMA.	3
II.2.1 <i>¿Qué es el cáncer de mama?</i>	3
II.2.2 <i>¿Qué es una mastografía?</i>	4
II.3 PROCESAMIENTO DIGITAL DE IMÁGENES	4
II.3.1 <i>Áreas de aplicación del procesamiento digital de imágenes</i>	5
II.3.2 <i>Adquisición de imágenes</i>	9
II.3.3 <i>Muestreo y cuantificación de imágenes</i>	10
II.4 OPERACIONES PARA EL PROCESAMIENTO DIGITAL DE IMÁGENES.	12
II.4.1 <i>Operaciones individuales</i>	12
II.4.2 <i>Operaciones de vecindad</i>	16
II.4.3 <i>Operaciones geométricas</i>	19
II.4.4 <i>Operaciones entre imágenes</i>	21
II.5 SOLUCIONES DE ALTO RENDIMIENTO PARA PROCESAMIENTO DIGITAL DE IMÁGENES	24
II.5.1 <i>FPGA (Field Programmable Gate Array)</i>	25
II.5.2 <i>DSP (Digital Signal Processor)</i>	27
II.6 FABRICANTES DE FPGAS	28
II.6.1 <i>Xilinx</i>	28

II.6.2 Altera	28
II.6.3 Lattice Semiconductor	28
II.6.4 QuickLogic	28
II.6.5 Atmel	29
II.6.6 Achronix Semiconductor	29
II.6.7 MathStar	29
II.7 FPGAS DE XILINX	29
II.7.1 Serie Virtex	29
II.7.2 Serie Spartan	31
III. IMPLEMENTACIÓN	33
III.1 INTRODUCCION	33
III.2 HARDWARE	33
III.2.1 Plataforma de evaluación ML403	33
III.3 HERRAMIENTAS DE DESARROLLO	40
III.3.1 EDK de Xilinx	40
III.4 DISEÑO DEL SISTEMA	41
III.4.1 Creación de la plataforma de hardware	41
III.4.2 Creación de la plataforma de software	44
III.4.3 Creación de la aplicación de software	48
III.4.4 Verificación del software (depuración)	49
IV. DISCUSIÓN DE RESULTADOS	50
IV.1 INTRODUCCIÓN	50
IV.2 COMPARACIÓN DE RESULTADOS	50
IV.3 CONCLUSIONES	51
V. CONCLUSIÓN	54
V.1 INTRODUCCIÓN	54
V.2 TRABAJO A FUTURO	54
BIBLIOGRAFÍA	56

ÍNDICE DE FIGURAS

Figura 1. Imagen de una mastografía (Microsoft Student 2008 [DVD], 2007c)	4
Figura 2. Fotografía digital producida en 1921. (Gonzalez, y otros, 2002)	5
Figura 3. Ejemplo de una imagen de rayos gamma, muestra el esqueleto del paciente. (Gonzalez, y otros, 2002)	6
Figura 4. Ejemplo de una imagen de rayos X, muestra el pecho de un paciente. (Gonzalez, y otros, 2002).....	6
Figura 5. Ejemplos de una imagen ultravioleta. De izquierda a derecha, (a) Imagen de un maíz normal (b) Imagen de un maíz infectado con “smut”. (Gonzalez, y otros, 2002).....	7
Figura 6. Ejemplos de una imagen de la luz de un microscopio. De izquierda a derecha, (a) Taxol (agente anti cancerígeno) aumentado 250 veces, (b) Colesterol aumentado 40 veces, (c) Microprocesador aumentado 60 veces. (Gonzalez, y otros, 2002).....	7
Figura 7. Ejemplo de una imagen multiespectral. Huracán Andrew tomada por NOAA GEOS (Geostationary Environmental Operational Satellite). (Gonzalez, y otros, 2002).....	8
Figura 8. Ejemplo de una imagen de radar, tomada a bordo de una nave espacial. Área montañosa al sureste del Tíbet. (Gonzalez, y otros, 2002).....	8
Figura 9. Ejemplo de una imagen de resonancia magnética de una rodilla humana. (Gonzalez, y otros, 2002).....	9
Figura 10. (a) Un sensor para obtener imágenes, (b) Una línea de sensores, (c) Una arreglo de sensores. (Gonzalez, y otros, 2002).....	10
Figura 11. Obtención de una imagen mediante un arreglo de sensores. (a) Imagen continua proyectada en un arreglo de sensores. (b) Resultado del muestreo y la cuantificación. (Gonzalez, y otros, 2002)	11
Figura 12. Convención de coordenadas utilizada para representar una imagen digital. (Gonzalez, y otros, 2002)	12
Figura 13. Operación individual. (Pajares, y otros, 2002).....	13
Figura 14. (a) Imagen original. (b) Imagen inversa o negativo de la imagen original. (Enema opaco. Enciclopedia Encarta, Herbert Wagner/Phototake NYC).....	13
Figura 15. (a) Imagen original. (b) Imagen obtenida al aplicar el operador umbral. (Enciclopedia Encarta, Omikron/Science Source/Photo Researches, Inc.).....	14
Figura 16. (a) Imagen original. (b) Imagen obtenida al aplicar el operador de corrimiento lineal con $p_1 = 35$. (c) Imagen obtenida con el operador de corrimiento lineal con $p_1 = -60$.	

(Tomografía axial computarizada, Enciclopedia Encarta, Hank Morgan/Science Source/Photo Researches, Inc.).....	14
Figura 17. Representación del vector <i>RANGO</i> (rectángulo morado), recorriendo el vector <i>LUT</i> . (Nájera, 2007)	15
Figura 18. De izquierda a derecha, (a) Imagen original. (b) Imagen obtenida al aplicar el algoritmo de lupa de intensidades.	16
Figura 19. (a) Imagen original. (b) Imagen ruidosa. (c) Imagen obtenida al aplicar el filtro de la media. (Enciclopedia Encarta, Omikron/Science Source/Photo Researches, Inc.)	17
Figura 20. (a) Imagen original. (b) Resultado del filtrado con gradiente normal L1. (c) Resultado del filtrado con gradiente normal L2. (Enema de doble contraste de colon, Enciclopedia Encarta, CNRI/Science Source/Photo Researches, Inc.)	18
Figura 21. Ante cualquier transformación geométrica, los pixeles de la rejilla transformada (líneas discontinuas) no tienen por qué coincidir con los de la rejilla destino o final. (Pajares, y otros, 2004).....	19
Figura 22. (a) Imagen original. (b) Imagen rotada 180°. (Enciclopedia Encarta, Omikron/Science Source/Photo Researches, Inc.)	20
Figura 23. (a) Imagen original. (b) Resultado de la reflexión horizontal. (Enciclopedia Encarta, Omikron/Science Source/Photo Researches, Inc.)	20
Figura 24. (a) Imagen original. (b) Resultado de la reflexión vertical. (Enciclopedia Encarta, Omikron/Science Source/Photo Researches, Inc.)	21
Figura 25. Representación simbólica de la transformación de dos imágenes en una. (Pajares, y otros, 2004).....	22
Figura 26. (a) Imagen original A. (b) Imagen original B. (c) Suma (A + B). (d) Resta (A - B). (Anatomía humana, Enciclopedia Encarta)	23
Figura 27. (a) Imagen original A. (b) Imagen original B. (c) AND lógico. (d) OR lógico. (Anatomía humana, Enciclopedia Encarta)	24
Figura 28. Estructura de un bloque de FPGA. (Xilinx, Inc., 2008a).....	25
Figura 29. Esquema de un CLB de alto nivel. (Xilinx, Inc., 2008a).....	26
Figura 30. Diagrama de bloques de la plataforma de evaluación ML403. (Xilinx, Inc., 2006)	34
Figura 31. Descripción detallada de los componentes de la plataforma de evaluación ML403 (Parte frontal). (Xilinx, Inc., 2006).....	35
Figura 32. Descripción detallada de los componentes de la plataforma de evaluación ML403 (Parte trasera). (Xilinx, Inc., 2006).....	36
Figura 33. Diagrama de bloques de los distintos periféricos, buses, y el procesador PowerPC, generado por la herramienta XPS de Xilinx.	42
Figura 34. Detalle de los distintos periféricos, buses y procesador, mostrando la interconexión entre los distintos periféricos.....	43
Figura 35. Estructura del MicroKernel de Xilinx (XMK).....	45
Figura 36. Configuración de la plataforma de software.	46
Figura 37. Configuración del Sistema Operativo y de las librerías.....	47
Figura 38. Controladores asignados a los distintos periféricos presentes en la plataforma de hardware.	47
Figura 39. Archivos de la aplicación de software.	49
Figura 40. Ejecución del sistema con procesador embebido PowerPC 405, mostrando una imagen a través de la salida VGA.	53

Capítulo I

I. INTRODUCCIÓN

I.1 INTRODUCCIÓN

El campo del procesamiento digital de imágenes ha crecido enormemente en los últimos años. El procesamiento digital de imágenes se está utilizando en muchas disciplinas dando solución a un gran número de problemas. La importancia del tema se puede comprobar con el gran número de aplicaciones que existen para diferentes áreas entre las cuales se incluyen, defensa, entretenimiento, sistemas de seguridad, identificación personal, imágenes medicas, entre otras.

Generalmente, el procesamiento digital de imágenes requiere un poder de cómputo intensivo, por lo cual se requieren equipos de alto rendimiento para agilizar las aplicaciones de procesamiento digital de imágenes. Por años se ha realizado investigación para desarrollar sistemas de alto rendimiento utilizando algunos avances en hardware, entre los cuales se encuentran, el procesamiento paralelo, los procesadores digitales de señales (DSPs por sus siglas en ingles), hardware de propósito específico (ASICs por sus siglas en ingles), hardware reconfigurable como los arreglos de bloques lógicos programables (FPGAs por sus siglas en ingles).

I.2 OBJETIVOS

Para el desarrollo de este trabajo de tesis nos planteamos como objetivo principal el aumentar la velocidad de procesamiento de una imagen de mastografía utilizando hardware reconfigurable como plataforma de desarrollo.

Entonces es necesario implementar diversas técnicas de procesamiento digital de imágenes en un arreglo de bloques lógicos programables (FPGA) de modo que el procesamiento de una imagen de mastografía se realice en corto tiempo comparado con el tiempo de procesamiento de una computadora personal.

I.3 RESULTADOS ESPERADOS

Con el objetivo principal planteado, se espera obtener como resultado un sistema de procesamiento de imágenes de mastografía implementado en un FPGA, el cual permita aumentar la velocidad de procesamiento de una imagen de mastografía.

I.4 ORGANIZACIÓN DE LA TESIS

Este trabajo de tesis se divide en los siguientes capítulos o apartados,

Dedicatoria, Agradecimientos y Resumen, donde se muestra un panorama general del este trabajo de tesis.

Tabla de Contenido, la cual muestra la estructura jerárquica de los distintos capítulos del trabajo de tesis.

Índice de figuras, el cual muestra una lista de todas las figuras o imágenes mostradas en los distintos capítulos del trabajo de tesis.

Capítulo I, en este capítulo se muestra el objetivo principal, los resultados esperados y la organización del trabajo de tesis.

Capítulo II, en este capítulo se muestra un panorama general del cáncer, del cáncer de mama, y de la mastografía, se expone una introducción a las imágenes digitales y a los distintos algoritmos de procesamiento digital de imágenes, se realiza un resumen de las distintas plataformas de alto rendimiento utilizadas para procesamiento digital de imágenes, entre los que destacan los FPGAs, y los DSPs.

Capítulo III, en este capítulo se detalla la plataforma de hardware, la plataforma de desarrollo y las herramientas utilizadas en la implementación del trabajo de tesis.

Capítulo IV, en este capítulo se realiza la discusión de los resultados obtenidos en la implementación de este trabajo de tesis.

Capítulo V, muestra las conclusiones y se propone trabajo a futuro para el mejoramiento de este trabajo de tesis.

Este trabajo de tesis contiene 40 figuras, 2 tablas, 23 ecuaciones, 22 referencias bibliográficas, todo esto en 56 páginas.

Capítulo II

II. MARCO TEÓRICO

II.1 INTRODUCCIÓN.

En este capítulo se muestra un panorama general del cáncer, del cáncer de mama, y de la mastografía, se expone una introducción a las imágenes digitales y a los distintos algoritmos de procesamiento digital de imágenes, se realiza un resumen de las distintas plataformas de alto rendimiento utilizadas para procesamiento digital de imágenes, entre los que destacan los FPGAs, y los DSPs.

II.2 CANCER DE MAMA.

Según estimaciones de la Organización Mundial de la Salud (OMS), en 2005 7,6 millones de personas murieron de cáncer en todo el mundo y se estima que unos 84 millones de personas morirán como consecuencia del cáncer en los próximos 10 años. El cáncer es la segunda causa de muerte en los adultos en el mundo occidental, después de las enfermedades cardiovasculares, y es la principal causa de muerte por enfermedad (por detrás de los accidentes de cualquier tipo) en niños de 1 a 14 años. Existe una gran variabilidad en la incidencia de los distintos tipos de cáncer según el área geográfica. Estudios sobre poblaciones que han emigrado de un área geográfica a otra diferente sugieren que tales variaciones se deben más a diferencias en el estilo de vida que al origen étnico. Esto es comprensible, si se considera que la mayor parte de los cánceres se relacionan con factores ambientales más que con la herencia, pese a que ambos influyen. (Microsoft Student 2008 [DVD], 2007a)

II.2.1 ¿Qué es el cáncer de mama?

El cáncer de mama es un tumor maligno en los tejidos glandulares de la mama. Estos tumores, también denominados carcinomas, aparecen cuando los procesos que controlan la reproducción celular normal se alteran permitiendo que una única célula anormal se multiplique con rapidez. Los carcinomas tienden a destruir una proporción creciente del tejido mamario normal y con el tiempo pueden diseminarse (metastatizar) a otras partes del cuerpo.

El cáncer de mama es la causa más frecuente de tumor maligno en la mujer, excluyendo el cáncer de piel. Es un problema de salud grave en muchas regiones del mundo y su frecuencia es mayor en los países industrializados. En los últimos años, como consecuencia del desarrollo de un programa de diagnóstico precoz realizado a gran escala en mujeres de alto riesgo, se han

detectado y tratado un número mayor de cánceres de mama en mujeres sin síntomas. El cáncer de mama puede afectar también al hombre, pero es 100 veces más frecuente en la mujer. (Microsoft Student 2008 [DVD], 2007b)

II.2.2 ¿Qué es una mastografía?

La mamografía o mastografía es una técnica especial de rayos X que se utiliza para visualizar los tejidos blandos de la mama como un medio de detección del cáncer de mama en la mujer. Esta mamografía mostrada en la **Figura 1**, muestra calcificaciones (manchas de alta densidad) en una neoplasia, el pezón está a la izquierda. La mayoría de los cánceres de mama se originan en el conducto de la glándula mamaria o glándula productora de leche. El resto procede del propio tejido glandular. La mayoría de los tumores demuestran una conducta invasiva (maligna) precoz, aunque también existen formas no invasivas. (Microsoft Student 2008 [DVD], 2007c)

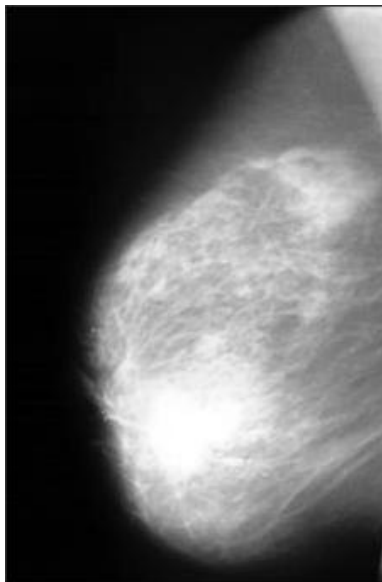


Figura 1. Imagen de una mastografía (Microsoft Student 2008 [DVD], 2007c)

La mastografía o mamografía ayuda al médico a detectar y valorar la presencia de anomalías en esta glándula. La mamografía se realiza de forma habitual en mujeres que carecen de síntomas de cáncer de mama. Esta técnica permite descubrir el cáncer en fases muy tempranas, cuando el tratamiento es más eficaz. La mamografía puede detectar anomalías de la mama de dimensiones tan reducidas como 0,5 cm, un tamaño demasiado pequeño para que una mujer o su médico sean capaces de percibir un bulto.

II.3 PROCESAMIENTO DIGITAL DE IMÁGENES

Iniciemos con la definición de una *imagen*, la cual se define como una función bidimensional $f(x, y)$, donde x, y son coordenadas espaciales, y la amplitud de f en cualquier par de coordenadas (x, y) es llamado intensidad de la imagen en ese punto. Cuando x, y toma valores finitos y los valores de amplitud de f son cantidades discretas, entonces a la imagen la llamamos *imagen digital*. (Gonzalez, y otros, 2002)

El campo del *procesamiento digital de imágenes* se refiere a procesar *imágenes digitales* mediante una computadora digital. Una imagen digital está compuesta por un número finito de elementos, cada uno con una localidad y valor particular, a los cuales comúnmente se le llaman píxeles y denotan los elementos de una imagen digital.

Una de las primeras aplicaciones de las imágenes digitales fue en la industria periodística, cuando las fotografías fueron enviadas por primera vez por cable entre Londres y Nueva York. Con la introducción del sistema de transmisión de fotografías por cable en los inicios de 1920 se redujo el tiempo de transporte requerido para transportar una fotografía a través del atlántico de más de una semana a menos de tres horas. La **Figura 2**, fue transmitida de esta forma y reproducida en una impresora telégrafo.



Figura 2. Fotografía digital producida en 1921. (Gonzalez, y otros, 2002)

Aunque, si bien el ejemplo citado anteriormente involucra imágenes digitales, no se considera procesamiento digital de imágenes, esto debido a que en la creación de esta imagen no se utilizó una computadora, la historia del procesamiento digital de imágenes está íntimamente ligada al desarrollo de la computadora digital.

II.3.1 Áreas de aplicación del procesamiento digital de imágenes

Las áreas de aplicación del procesamiento digital de imágenes son muy variadas, uno de los métodos más simples para desarrollar un entendimiento básico de lo extenso de las aplicaciones de procesamiento de imágenes es categorizarlas de acuerdo a su fuente. La principal fuente de energía para las imágenes hoy en día es el espectro de energía electromagnética. Otra fuente de energía importante incluye la acústica, electrónica (en la forma de rayos de electrones usados en la microscopía de electrones), entre otras.

II.3.1.1 Imágenes de rayos gamma

El mayor uso de las imágenes basadas en rayos gamma incluye la medicina nuclear y las observaciones astronómicas. En medicina nuclear, el enfoque es inyectar a un paciente un isótopo radioactivo que emita rayos gamma, las imágenes son creadas de las emisiones recolectadas por los detectores de rayos gamma. **Figura 3** muestra una imagen generada con rayos gamma.

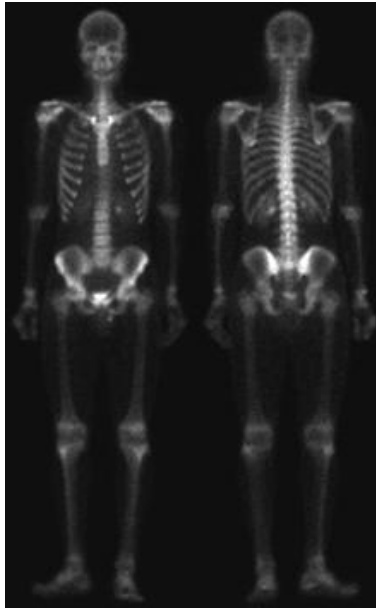


Figura 3. Ejemplo de una imagen de rayos gamma, muestra el esqueleto del paciente. (Gonzalez, y otros, 2002)

II.3.1.2 Imágenes de rayos X

Los rayos X están entre las fuentes de radiación electromagnética más antiguas utilizadas para generación de imágenes. El mejor uso conocido de los rayos X es el diagnóstico médico, pero también son usados en la industria y astronomía. En la **Figura 4** se muestra una imagen de rayos X del pecho de un paciente. La imagen se genera colocando al paciente entre la fuente de rayos X y una película sensible a la energía de los rayos X.



Figura 4. Ejemplo de una imagen de rayos X, muestra el pecho de un paciente. (Gonzalez, y otros, 2002)

II.3.1.3 Imágenes en la banda ultravioleta

Las aplicaciones de luz ultravioleta son variadas, entre las cuales se incluyen litografía, inspección industrial, microscopía, laser, imágenes biológicas, y observación astronómica. La luz ultravioleta es usada en la microscopia fluorescente, la cual es una de las áreas con más

crecimiento. La fluorescencia es un fenómeno descubierto en la mitad del siglo XIX, cuando se observó que el mineral fluorita reaccionaba cuando la luz ultravioleta se colocaba directamente sobre este. La **Figura 5(a)** muestra el resultado de la capacidad de un microscopio fluorescente, dicha imagen muestra la imagen de un maíz normal, la **Figura 5(b)** muestra un maíz infectado con “smut” una enfermedad de los cereales.

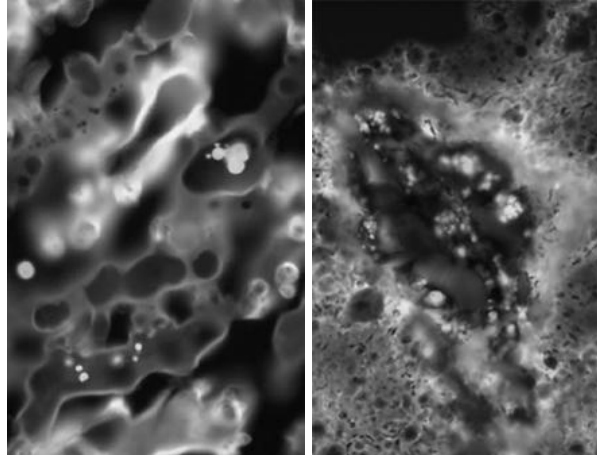


Figura 5. Ejemplos de una imagen ultravioleta. De izquierda a derecha, (a) Imagen de un maíz normal (b) Imagen de un maíz infectado con “smut”. (Gonzalez, y otros, 2002)

II.3.1.4 Imágenes en la banda visible e infrarroja

Considerando que la banda visible del espectro electromagnético es la más familiar en nuestras actividades, no es de sorprenderse que las imágenes en esta banda pesen más que las otras en términos de alcance. La banda infrarroja comúnmente es usada en conjunto con la banda visible, por eso se han agrupado las dos bandas solo para efectos ilustrativos. La **Figura 6** muestra tres ejemplos de imágenes obtenidas con la luz de un microscopio.

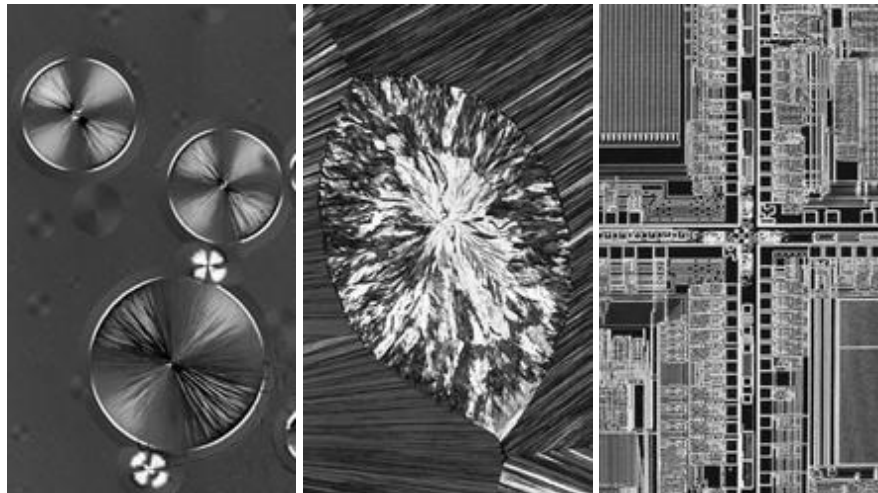


Figura 6. Ejemplos de una imagen de la luz de un microscopio. De izquierda a derecha, (a) Taxol (agente anti cancerígeno) aumentado 250 veces, (b) Colesterol aumentado 40 veces, (c) Microprocesador aumentado 60 veces. (Gonzalez, y otros, 2002)

La observación y predicción del clima son aplicaciones de imágenes satelitales multispectrales. A continuación se muestra la imagen de un huracán tomada por el NOAA (National Oceanographic and Atmospheric Administration), el satélite utilizó sensores en la banda visible e infrarroja. El ojo del huracán es claramente visible en la **Figura 7**.



Figura 7. Ejemplo de una imagen multispectral. Huracán Andrew tomada por NOAA GEOS (Geostationary Environmental Operational Satellite). (Gonzalez, y otros, 2002)

II.3.1.5 Imágenes en la banda de microondas

La aplicación dominante en la banda de microondas es el radar. La única característica de las imágenes de radar es la habilidad de coleccionar datos sobre cualquier región virtual a cualquier hora, sin importar el clima o las condiciones de luz en el ambiente, algunas ondas de radar pueden penetrar nubes y en algunas circunstancias pueden ver a través de la vegetación, hielo, e incluso a través de arena seca. En algunos casos el radar es la única manera de explorar algunas regiones de la superficie de la tierra. La **Figura 8**, muestra un área montañosa al sur del Tíbet, aproximadamente a 90Km al este de la ciudad de Lhasa, la imagen fue tomada por un radar a bordo de una nave espacial. Nótese la claridad y el detalle de la imagen sin interferencia de nubes o alguna otra condición atmosférica que normalmente aparecen en las imágenes tomadas en la banda visible.

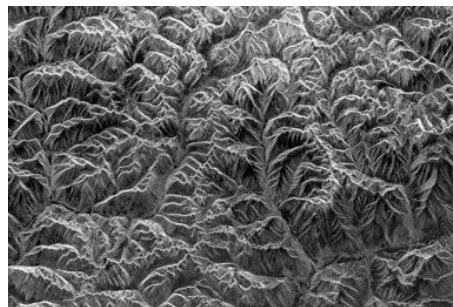


Figura 8. Ejemplo de una imagen de radar, tomada a bordo de una nave espacial. Área montañosa al sureste del Tíbet. (Gonzalez, y otros, 2002)

II.3.1.6 Imágenes en la banda de radio

Las aplicaciones de las imágenes en la banda de radio son en medicina y astronomía. En medicina las ondas de radio son usadas en imágenes de resonancia magnética. Esta técnica coloca un paciente en un poderoso imán y pasa ondas de radio a través de su cuerpo en pequeños pulsos. Cada pulso causa en respuesta un pulso de ondas de radio emitidas por los tejidos del paciente. La localización de donde estas señales se originan y su intensidad son determinadas por una computadora, la cual produce una imagen en dos dimensiones de una sección del paciente, esta técnica puede producir imágenes en cualquier lugar. En la **Figura 9**, se muestra la imagen de una rodilla utilizando la técnica de resonancia magnética.



Figura 9. Ejemplo de una imagen de resonancia magnética de una rodilla humana. (Gonzalez, y otros, 2002)

II.3.2 Adquisición de imágenes

El tipo de imágenes en las que estamos interesados son generadas por la combinación de una fuente de “iluminación” y la reflexión o absorción de energía de los elementos en la “escena”. Por ejemplo, la iluminación se puede originar desde una fuente de energía electromagnética como la de radar, infrarroja, o de rayos X, aunque también puede ser generada desde una fuente menos tradicional como el ultrasonido. De forma similar los elementos de una escena pueden ser objetos familiares, pero también pueden ser moléculas, formaciones de roca enterradas, o un cerebro humano.

Dependiendo de la naturaleza de la fuente, la energía de iluminación es reflejada o transmitida a través de los objetos. En algunas aplicaciones, la energía reflejada o transmitida es concentrada en un foto convertidor, el cual convierte la energía en luz visible. La **Figura 10** muestra los tres principales arreglos de sensores utilizados para transformar la energía de iluminación en imágenes digitales. La idea es simple, la energía entrante es convertida en voltaje por la combinación de energía eléctrica de entrada y el resultado del material sensible que responde a un tipo particular de energía. La forma de onda del voltaje de salida es la respuesta de los sensores.

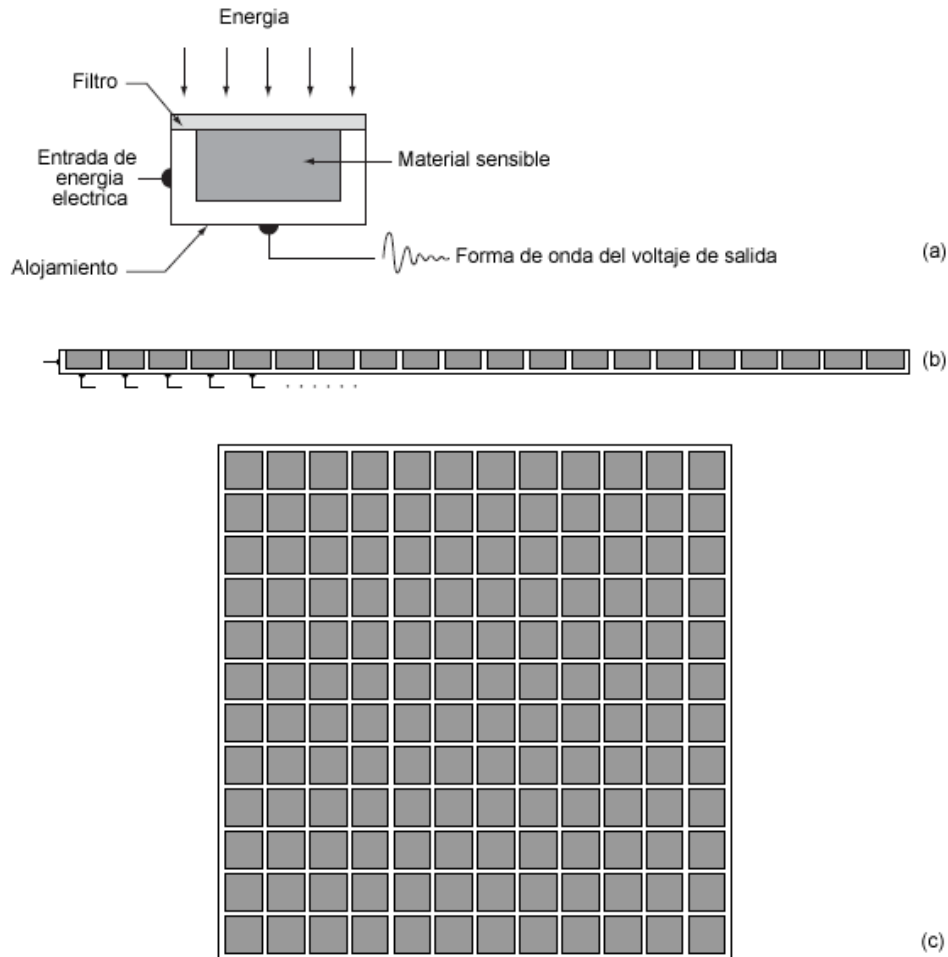


Figura 10. (a) Un sensor para obtener imágenes, (b) Una línea de sensores, (c) Una arreglo de sensores. (Gonzalez, y otros, 2002)

II.3.3 Muestreo y cuantificación de imágenes

De la discusión en la sección anterior, podemos ver que existen numerosas formas de adquirir imágenes, pero el objetivo en todas las formas es el mismo: generar imágenes digitales a partir de datos censados. La salida de la mayoría de los sensores es una onda de voltaje continua en la cual su amplitud y su comportamiento espacial están relacionados al fenómeno físico que se está censando. Para crear una imagen digital se necesita convertir los datos continuamente censados en formato digital. Esto involucra dos procesos: *muestreo* y *cuantificación*.

II.3.3.1 Conceptos básicos en muestreo y cuantificación

Digitalizar los valores de amplitud es llamado *cuantificación*. En la práctica, el método de muestreo es determinado por el arreglo de sensores utilizados para generar la imagen. Cuando una imagen es generada por un solo sensor combinado con movimientos mecánicos, no hay límites en que tan finamente se puede hacer la muestra de la imagen debido a que los movimientos mecánicos pueden ser muy exactos. Sin embargo los límites prácticos son establecidos por las imperfecciones en la óptica utilizada.

Cuando se utiliza una línea de sensores para adquirir la imagen, el número de sensores en la línea establece las limitaciones del muestreo en una dirección de la imagen. El movimiento mecánico en la otra dirección de la imagen puede controlarse con mayor exactitud, la cuantificación de la salida de los sensores completa el proceso en la generación de la imagen digital.

Cuando un arreglo de sensores es utilizado para la adquisición de la imagen, no hay movimiento mecánico y el número de sensores en el arreglo establece el límite de muestreo en ambas direcciones. La **Figura 11** muestra este concepto. La **Figura 11(a)** muestra una imagen continua proyectada en el plano de un arreglo de sensores. La **Figura 11(b)** muestra la imagen después del muestreo y la cuantificación.

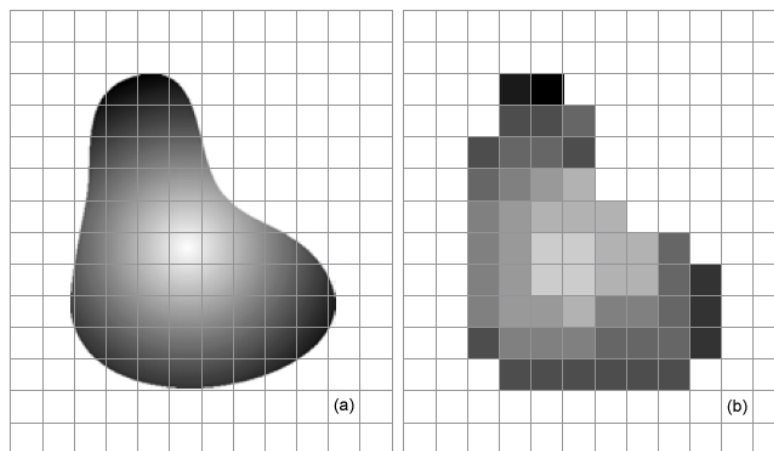


Figura 11. Obtención de una imagen mediante un arreglo de sensores. (a) Imagen continua proyectada en un arreglo de sensores. (b) Resultado del muestreo y la cuantificación. (Gonzalez, y otros, 2002)

II.3.3.2 Representación de una imagen digital

El resultado del muestreo y la cuantificación es una matriz de números reales. Se asume que el muestreo de una imagen $f(x, y)$ da como resultado una imagen digital con M filas y N columnas. Los valores de las coordenadas (x, y) ahora son valores discretos. Por claridad y conveniencia en la notación, se utilizan valores enteros para estas coordenadas discretas. Así, los valores de la coordenada en el origen es $(x, y) = (0, 0)$. La **Figura 12** muestra la convención de las coordenadas utilizada para representar una imagen digital.

La notación introducida en el párrafo anterior permite escribir una imagen digital completa de dimensiones $M \times N$ en la siguiente matriz:

$$f(x, y) = \begin{bmatrix} f(0, 0) & f(0, 1) & \cdots & f(0, N - 1) \\ f(1, 0) & f(1, 1) & \cdots & f(1, N - 1) \\ \vdots & \vdots & \ddots & \vdots \\ f(M - 1, 0) & f(M - 1, 1) & \cdots & f(M - 1, N - 1) \end{bmatrix} \quad (\text{Ecuación 1})$$

El lado derecho de la ecuación es por definición una imagen digital. Cada elemento de la matriz es llamado *elemento de la imagen*, *elemento de la pintura*, *pixel* o *pel*.

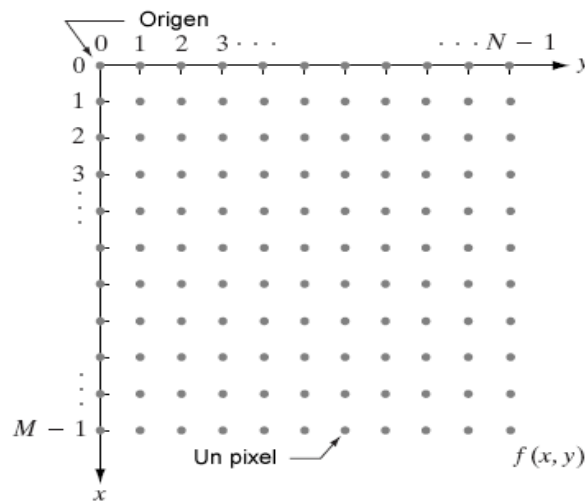


Figura 12. Convención de coordenadas utilizada para representar una imagen digital. (Gonzalez, y otros, 2002)

II.4 OPERACIONES PARA EL PROCESAMIENTO DIGITAL DE IMÁGENES.

II.4.1 Operaciones individuales

Las operaciones individuales implican la generación de una nueva imagen modificando el valor del pixel en una simple localización basándose en una regla global aplicada a cada localización de la imagen original. El proceso consiste en obtener el valor del pixel de una localización dada en la imagen, modificándolo por una operación lineal o no lineal y colocando el valor del nuevo pixel en la correspondiente localización de la nueva imagen. El proceso se repite para todas y cada una de las localizaciones de los píxeles en la imagen original. (Pajares, y otros, 2002)

En la **Figura 13**, se muestra un operador individual con una transformación uno a uno. El operador f se aplica a cada pixel en la imagen o sección de la imagen y la salida depende únicamente de la magnitud del correspondiente pixel de entrada; la salida es independiente de los píxeles adyacentes. La función transforma el valor del nivel de gris de cada pixel en la imagen y el nuevo valor se obtiene a través de la ecuación:

$$q(x, y) = f(p(x, y)) \quad (\text{Ecuación 2})$$

La función f puede ser un operador lineal o no lineal. La imagen resultante es de la misma dimensión que la original.

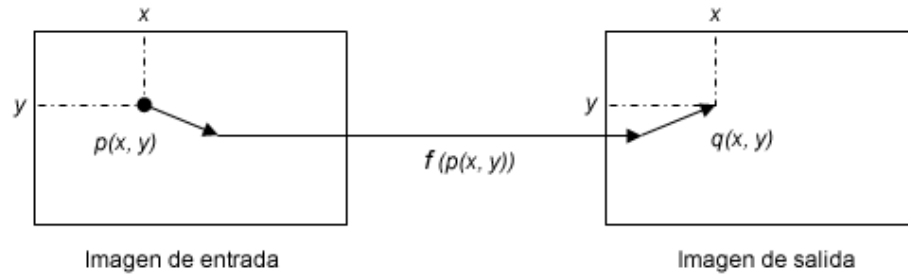


Figura 13. Operación individual. (Pajares, y otros, 2002)

II.4.1.1 Operador inverso o negativo

Este operador crea una imagen de salida que es la inversa de la imagen de entrada. Para una imagen con valores de gris en el rango de 0 a 255 la función de transformación resulta ser:

$$q = 255 - p \quad (\text{Ecuación 3})$$

Donde p es un pixel de la imagen original y q el pixel correspondiente a la imagen de salida. En la **Figura 14**, se ilustra la función de transformación dada por la ecuación anterior.

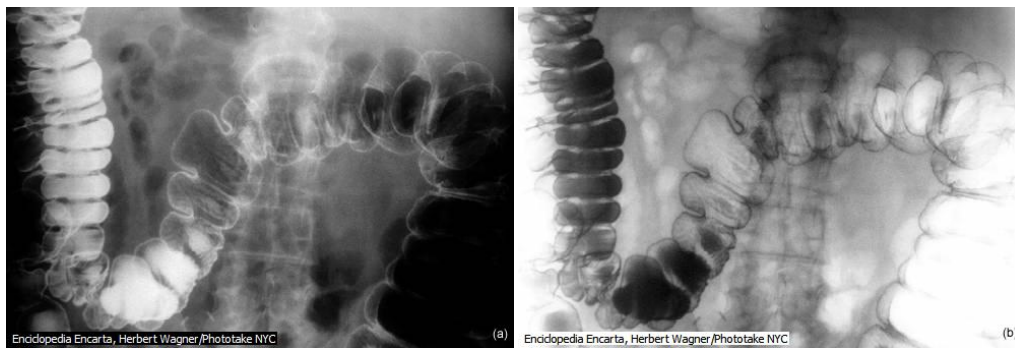


Figura 14. (a) Imagen original. (b) Imagen inversa o negativo de la imagen original. (Enema opaco. Enciclopedia Encarta, Herbert Wagner/Phototake NYC)

II.4.1.2 Operador umbral (binarización)

Esta clase de transformación crea una imagen de salida binaria a partir de una imagen de grises, donde el nivel de transición está dado por el parámetro de entrada p_1 . La función de transformación es la siguiente:

$$q = \begin{cases} 0 & \text{para } p \leq p_1 \\ 255 & \text{para } p > p_1 \end{cases} \quad (\text{Ecuación 4})$$

Donde p es un pixel de la imagen original, q el pixel correspondiente a la imagen de salida y p_1 el parámetro de entrada. En la **Figura 15(b)** se muestra la imagen obtenida al aplicar la transformación anterior a la imagen de la **Figura 15(a)** con $p_1 = 100$.

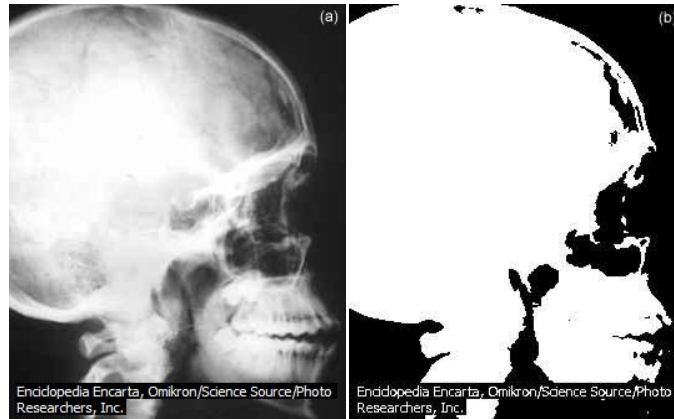


Figura 15. (a) Imagen original. (b) Imagen obtenida al aplicar el operador umbral.
(Enciclopedia Encarta, Omikron/Science Source/Photo Researches, Inc.)

II.4.1.3 Operador de corrimiento lineal

Esta clase de transformación crea una imagen de salida más clara o más oscura, donde el nivel de transición esta dado por el parámetro de entrada p_1 . La función de transformación es la siguiente:

$$q = p + p_1 \quad \text{si} \begin{cases} p_1 > 0 & \text{Aclara} \\ p_1 < 0 & \text{Obscurece} \end{cases} \quad (\text{Ecuación 5})$$

Donde p es un pixel de la imagen original, q el pixel correspondiente a la imagen de salida y p_1 el parámetro de entrada. En la **Figura 16**, se muestra la imagen obtenida al aplicar la transformación anterior, con distintos valores de p_1 .



Figura 16. (a) Imagen original. (b) Imagen obtenida al aplicar el operador de corrimiento lineal con $p_1 = 35$. (c) Imagen obtenida con el operador de corrimiento lineal con $p_1 = -60$.
(Tomografía axial computarizada, Enciclopedia Encarta, Hank Morgan/Science Source/Photo Researches, Inc.)

II.4.1.4 Lupa de intensidades

Este algoritmo tiene como objetivo resaltar aquellos puntos o áreas que no se logran distinguir a simple vista, ya que por la naturaleza de las imágenes de mastografía, estas son oscuras en ciertas regiones de interés. (Nájera, 2007)

Se sabe que el ojo humano no puede distinguir la diferencia de tonos de gris cuando estos tienen una diferencia entre sí muy pequeña. Tomando esto en cuenta se propone el siguiente procedimiento.

Se toman dos números entre 2 y 255, se toman en cuenta todos los valores que estén dentro del rango, estos valores serán ajustados por un nuevo valor de intensidad proporcional al tamaño del rango que se haya elegido.

$$f(x) = \psi x \quad (\text{Ecuación 6})$$

Para obtener ψ se debe conocer el rango, el cual se obtiene de la siguiente forma:

$$\psi = \frac{\Delta}{r_2 - r_1} \quad (\text{Ecuación 7})$$

De la ecuación anterior se obtiene lo siguiente:

$$f(x) = \frac{\Delta x}{r_2 - r_1} \quad (\text{Ecuación 8})$$

Donde r_1 es el límite inferior del rango y r_2 es el límite superior del rango, es decir si el rango es h entonces $r_1 = h - 1$ y $r_2 = h + 1$, con esto se obtienen los nuevos valores para cada píxel de la imagen. Así se obtiene lo siguiente:

$$\text{vector}[x] = f(x) \quad (\text{Ecuación 9})$$

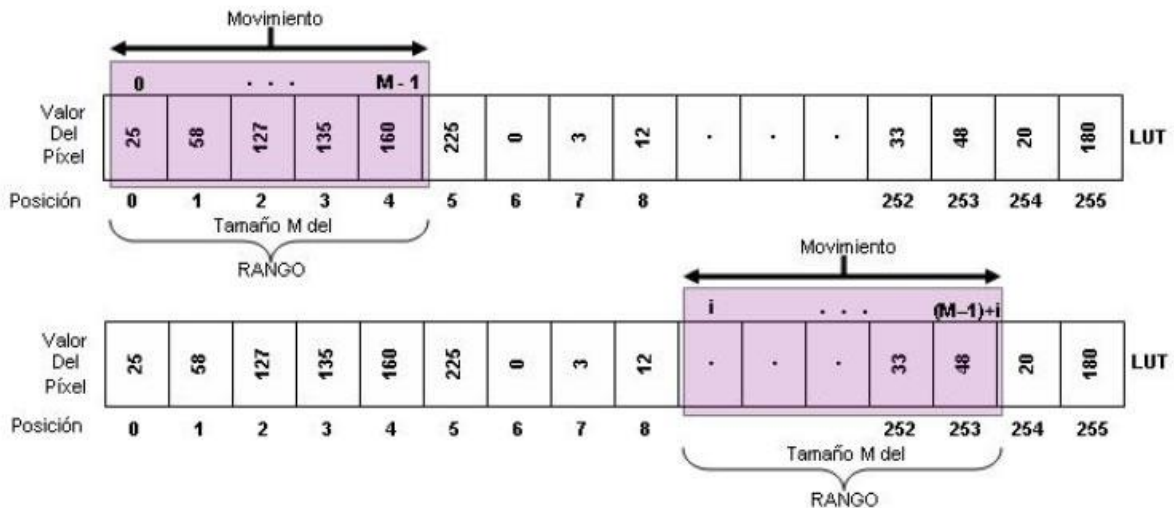


Figura 17. Representación del vector *RANGO* (rectángulo morado), recorriendo el vector *LUT*. (Nájera, 2007)

Como se puede apreciar en la **Figura 17**, el rectángulo de color morado representa el vector $\text{vector}[x]$ de tamaño *RANGO*, el cual procesara solo el espacio que se intercepta con el vector

LUT, este proceso se aplica a toda la imagen, es decir cada vez que se mueva el rectángulo morado, el vector *LUT* actualiza su punto inicial y final ya sea sumando o restando el número de pasos que se haya desplazado.

Debido a que la imagen se procesa por rangos, es de suponer que existen áreas que no sufren cambios, estas áreas no procesadas se muestran en color amarillo en la imagen resultante. En la **Figura 18** se muestra el resultado de aplicar el algoritmo de lupa de intensidades.

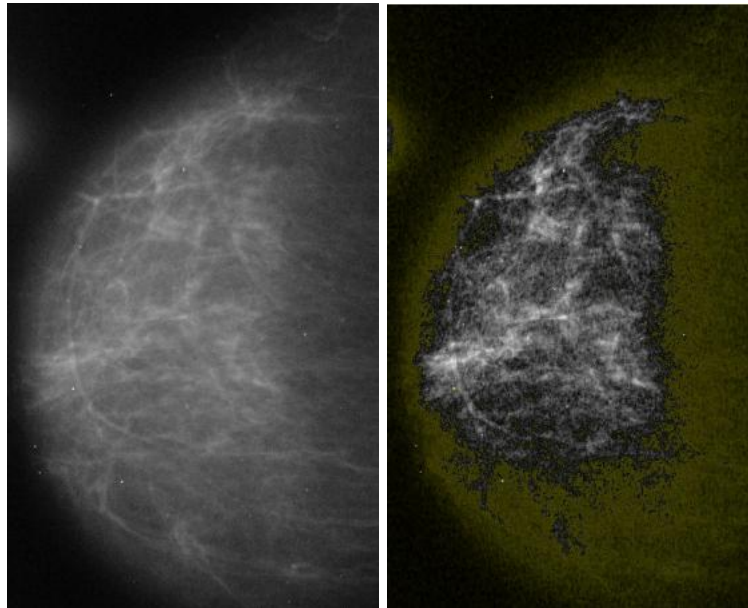


Figura 18. De izquierda a derecha, (a) Imagen original. (b) Imagen obtenida al aplicar el algoritmo de lupa de intensidades.

II.4.2 Operaciones de vecindad

Las operaciones de vecindad utilizan el mismo procedimiento que las operaciones individuales, excepto que el nuevo valor del pixel en la imagen de salida, depende de una combinación de los valores de los pixeles en la vecindad del pixel de la imagen original.

Básicamente consiste en transformar el valor de un pixel p en la posición (x, y) teniendo en cuenta los valores de los pixeles vecinos. La imagen de salida será de dimensión inferior a la original, ya que esta operación no se puede realizar sobre los pixeles extremos de la imagen original al no tener todos los vecinos.

A continuación se describen las distintas operaciones de vecindad utilizadas para la realización de este proyecto.

II.4.2.1 Filtro de la media

También denominado filtro de suavizado, se trata de una técnica aplicada directamente en el dominio espacial. Dada una imagen $I(x, y)$, se obtiene una imagen suavizada $I'(x, y)$ cuya intensidad para cada punto (x, y) se calcula promediando los valores de intensidad de los

pixeles de la imagen I incluidos en el entorno de vecindad predefinido de (x, y) . La operación puede sintetizarse de la siguiente expresión:

$$I'(x, y) = \frac{1}{P} \sum_{(i,j) \in S} I(i, j) \quad (\text{Ecuación 10})$$

Para todos los pixeles (x, y) de $I(x, y)$, S es el conjunto de coordenadas de los puntos situados en el entorno de vecindad de (x, y) , incluido el propio (x, y) , y P es el número total de puntos del entorno de vecindad. En la **Figura 19**, se muestra la imagen obtenida al aplicar el filtro anterior.

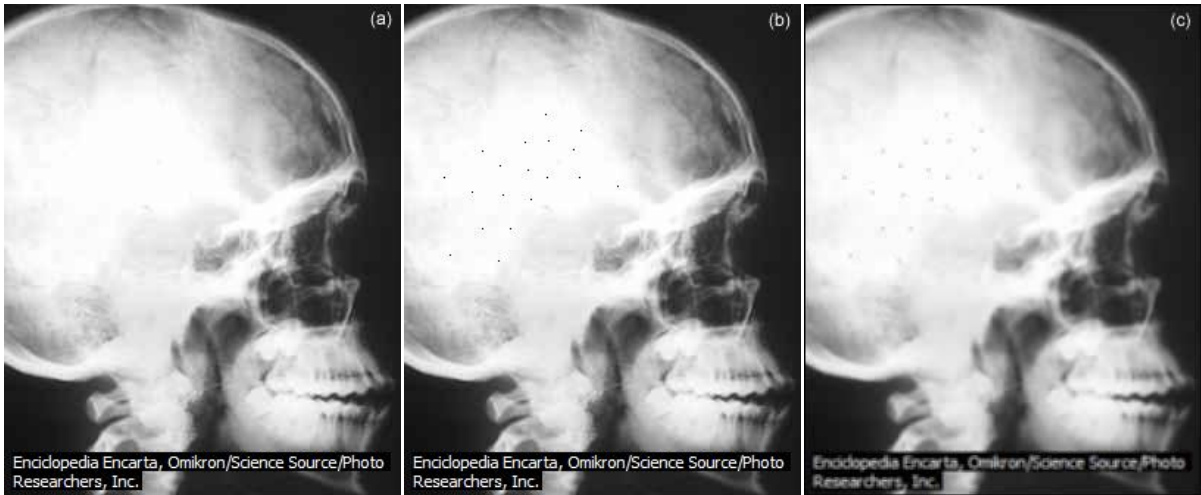


Figura 19. (a) Imagen original. (b) Imagen ruidosa. (c) Imagen obtenida al aplicar el filtro de la media. (Enciclopedia Encarta, Omikron/Science Source/Photo Researches, Inc.)

II.4.2.2 Filtro de extracción de bordes

Los puntos de borde, o simplemente bordes son pixeles alrededor de los cuales la imagen presenta una brusca variación en los niveles de gris. El objetivo consiste en dada una imagen, que puede o no estar corrompida por ruido, localizar los bordes más probables generados por elementos de la escena y no por ruido.

II.4.2.2.1 Gradiente de una imagen

El gradiente es una operación que determina la dirección de máximo crecimiento de una función, en el caso de dos dimensiones y en su versión continua se define como una operación que se aplica a una función de dos variables $f(x, y)$. (Martin, 2008a) Su forma matemática es:

$$\nabla f(x, y) = \hat{i} \frac{\partial f(x, y)}{\partial x} + \hat{j} \frac{\partial f(x, y)}{\partial y} \quad (\text{Ecuación 11})$$

Donde \hat{i}, \hat{j} son los vectores unitarios en las direcciones x e y . Si quisiéramos aplicar esta forma de la derivada para una función bidimensional, como es el caso de una imagen, en particular para una imagen digital nos enfrentamos al problema de que los pixeles representan un valor

escalar y no vectorial, por lo cual se debe hacer una simplificación de la expresión anterior. Utilizando la norma $L2$ para la magnitud del vector tendremos:

$$\nabla I[x, y] = \sqrt{\left(\frac{\delta I}{\delta x}\right)^2 + \left(\frac{\delta I}{\delta y}\right)^2} \quad (\text{Ecuación 12})$$

Lo cual se puede escribir como:

$$\nabla I[x, y] = \sqrt{(I[x + 1, y] - I[x, y])^2 + (I[x, y + 1] - I[x, y])^2} \quad (\text{Ecuación 13})$$

Pero como la diferencia máxima que podemos encontrar en una imagen de L bits por pixel en alguno de sus canales es $(2^L - 1)$, entonces podemos tener valores fuera de rango hasta por factor de dos, ya que cada termino puede tomar ese valor. Para evitar este problema debemos acotar la salida, de donde la forma anterior adaptada a un ambiente digital puede escribirse de la siguiente manera:

$$\nabla I[x, y] = \frac{1}{2} \sqrt{(I[x + 1, y] - I[x, y])^2 + (I[x, y + 1] - I[x, y])^2} \quad (\text{Ecuación 14})$$

Un método menos ávido de operaciones numéricas de punto flotante es utilizar la norma $L1$ o de cuerdas, en esta se toman las diferencias completas en las direcciones x e y como si se caminase en una ciudad con manzanas cuadradas e iguales y la regla es que solo se puede caminar sobre las calles sin ingresar a las manzanas en diagonal. Usando esta norma tendremos que el gradiente digital en norma $L1$ tomara la siguiente forma:

$$\nabla I[x, y] = \frac{1}{2} \{|I[x + 1, y] - I[x, y]| + |I[x, y + 1] - I[x, y]|\} \quad (\text{Ecuación 15})$$

En la **Figura 20** se muestra el efecto del gradiente en norma $L1$ y gradiente norma $L2$ sobre una imagen.

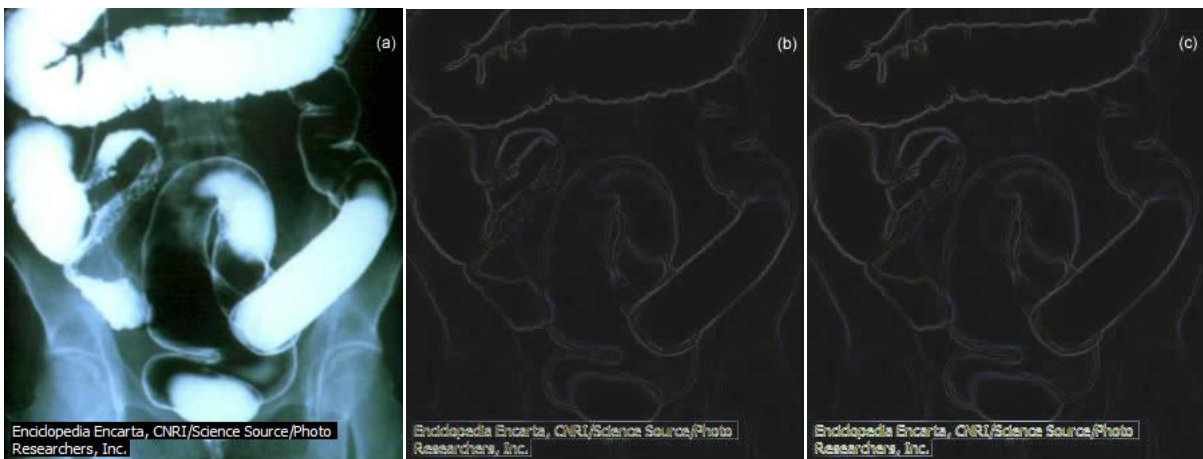


Figura 20. (a) Imagen original. (b) Resultado del filtrado con gradiente normal L1. (c) Resultado del filtrado con gradiente normal L2. (Enema de doble contraste de colon, Enciclopedia Encarta, CNRI/Science Source/Photo Researches, Inc.)

II.4.3 Operaciones geométricas

A menudo, para el análisis de las escenas representadas en las imágenes, queremos investigar más específicamente un área dentro de dicha imagen, llamada *región de interés*. Para hacer esto, necesitamos operaciones geométricas que modifiquen las coordenadas espaciales de la imagen. El objetivo fundamental de una operación geométrica es la transformación de los valores de una imagen, tal y como podría observarse desde otro punto de vista. Así, las operaciones de magnificar o reducir una imagen no son otra cosa que aproximar o alejar el punto de vista; rotarla equivale a girar el punto de observación y trasladarla es hacer lo propio con dicho punto. (Pajares, y otros, 2004)

En la **Figura 21**, se muestra una transformación geométrica, que en este caso concreto resulta ser una rotación. La rejilla continua muestra la disposición convencional de píxeles de la imagen original, mientras que la rejilla discontinua ilustra cómo queda la rejilla original después de girarla un ángulo dado alrededor de un punto determinado, es decir después de rotarla.

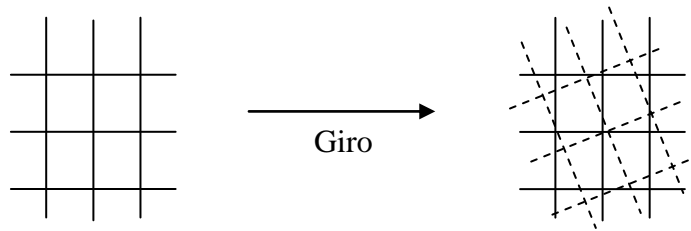


Figura 21. Ante cualquier transformación geométrica, los píxeles de la rejilla transformada (líneas discontinuas) no tienen por qué coincidir con los de la rejilla destino o final. (Pajares, y otros, 2004)

Por consiguiente, para llegar al resultado final es preciso:

- Determinar las coordenadas de cada píxel (x, y) perteneciente a la rejilla original en la rejilla transformada (en líneas discontinuas). En general, estos nuevos píxeles (x', y') obtenidos tras la transformación no serán valores enteros y, por tanto, no coincidirán con píxeles en la rejilla destino.
- Calcular los valores de los píxeles (x, y) finales en la rejilla destino a partir de los valores conocidos de píxeles (x', y') .

II.4.3.1 Rotación simple de 180°

En la rotación simple de 180° es claro que las dimensiones de la imagen no se modifican, el número de columnas y renglones sigue siendo el mismo. Y lo que ha sucedido es un cambio en la posición de los píxeles de tal manera que los índices de los renglones y las columnas se han invertido. (Martín, 2008b) La relación entre la imagen original $I[x, y]$ y la transformada $I'[x', y']$ se puede mostrar que es:

$$I'[x', y'] = I[N - 1 - x, M - 1 - y] \quad (\text{Ecuación 16})$$

Donde M corresponde al número filas en la imagen y N al número de columnas.



Figura 22. (a) Imagen original. (b) Imagen rotada 180°. (Enciclopedia Encarta, Omikron/Science Source/Photo Researches, Inc.)

II.4.3.2 Reflexión horizontal

Esta operación consiste en intercambiar las columnas de una imagen, lo cual se representa de la siguiente forma:

$$I'[x', y'] = I[N - 1 - x, y] \quad (\text{Ecuación 17})$$

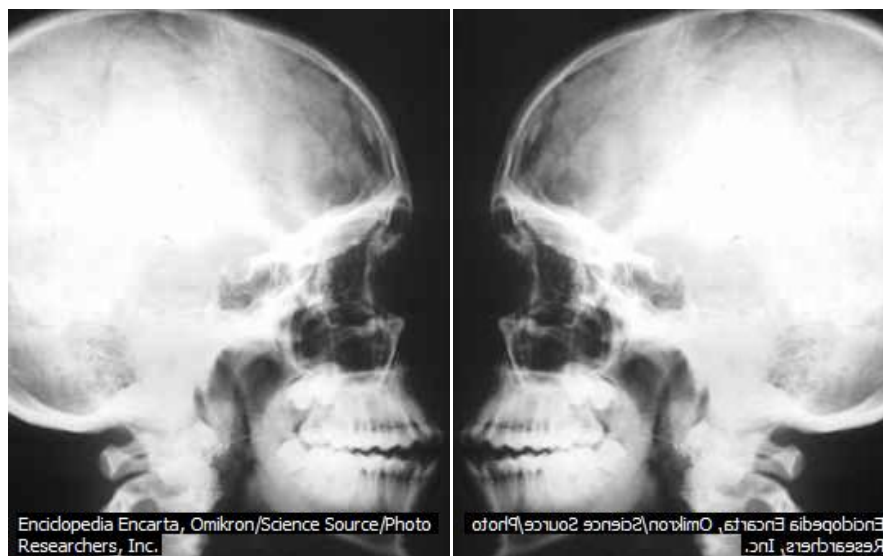


Figura 23. (a) Imagen original. (b) Resultado de la reflexión horizontal. (Enciclopedia Encarta, Omikron/Science Source/Photo Researches, Inc.)

II.4.3.3 Reflexión vertical

Esta operación consiste en intercambiar los renglones de una imagen, lo cual se representa de la siguiente forma:

$$I'[x',y'] = I[x,M - 1 - y] \quad (\text{Ecuación 18})$$

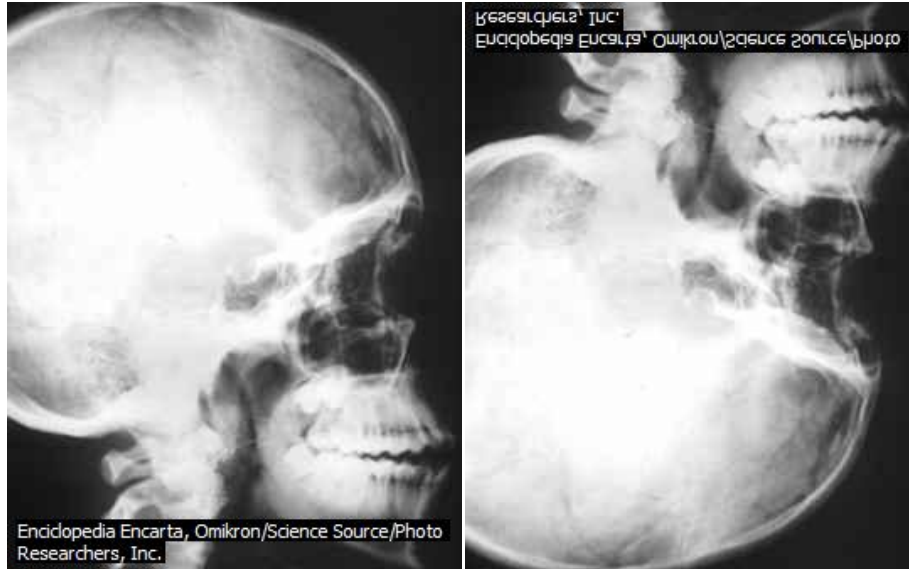


Figura 24. (a) Imagen original. (b) Resultado de la reflexión vertical. (Enciclopedia Encarta, Omikron/Science Source/Photo Researches, Inc.)

II.4.4 Operaciones entre imágenes

Este tipo de transformaciones utilizan la información contenida en la misma localización (posición de los píxeles) de dos imágenes de entrada A y B para crear una nueva imagen C . La dimensión de las imágenes es la misma. La función de la transformación f_D puede ser lineal o no. Esto es, la información en una localización de píxel de una imagen se combina con la información de la correspondiente localización de la segunda imagen para obtener el valor, también en la misma localización de píxel de la tercera imagen (Pajares, y otros, 2004). La función característica está dada por la siguiente ecuación:

$$C_{x,y} = f_D(A_{x,y}, B_{x,y}) \quad (\text{Ecuación 19})$$

Donde f_D es una función de dos variables y los subíndices x e y de 0 a M y de 0 a N respectivamente (que son las dimensiones de las imágenes). La función f_D puede ser adición, sustracción, división, exponenciación, máximo o cualquier otra función que se pueda definir. La función deberá tener un factor de escala apropiado k para mantener los valores de salida dentro del rango adecuado, así como para evitar desbordamientos y valores negativos.

Este proceso se representa de la siguiente forma:

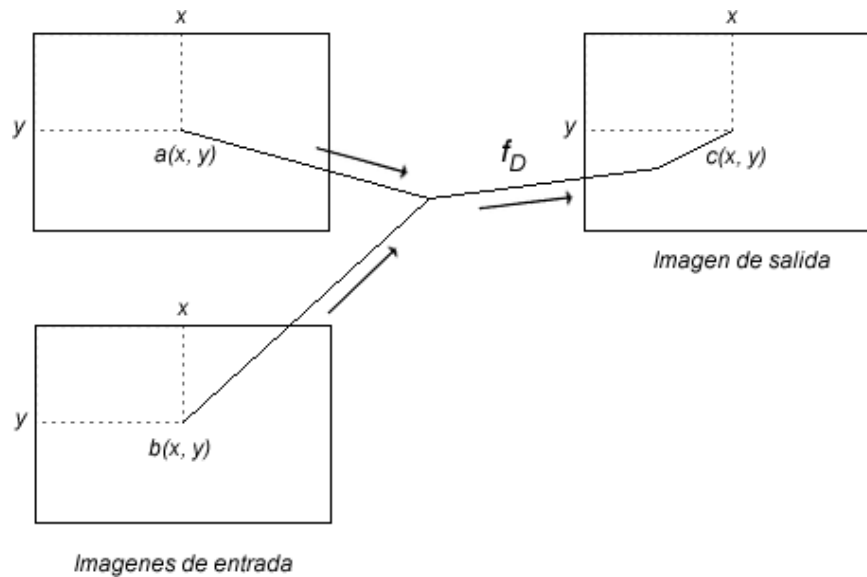


Figura 25. Representación simbólica de la transformación de dos imágenes en una. (Pajares, y otros, 2004)

II.4.4.1 Operaciones aritméticas

La suma puede utilizarse para reducir los efectos del ruido en la imagen. (Pajares, y otros, 2002) El valor de la salida está dado por:

$$C_{x,y} = \frac{(A_{x,y} + B_{x,y})}{k} \quad (\text{Ecuación 20})$$

Donde k es 2 para el caso de dos imágenes de entrada. Si la suma abarcase más de dos imágenes k tomaría el valor del número de imágenes utilizadas. Los valores de salida finales deben redondearse por defecto o por exceso.

La resta de imágenes es una técnica útil para detectar el cambio producido en dos imágenes que han sido captadas en dos instantes de tiempo diferentes. Los datos de ambas imágenes también pueden representar pérdidas de calor o frío si la fuente de datos es el espectro infrarrojo. Puesto que en el procesamiento de imágenes se utilizan números positivos, es necesario definir la salida de alguna manera que haga todos los valores positivos. Para este caso se utiliza el valor absoluto de la diferencia, la relación está dada por:

$$C_{x,y} = |A_{x,y} - B_{x,y}| \quad (\text{Ecuación 21})$$

La **Figura 26** muestra el resultado de aplicar la suma, y la resta entre la imagen original A y la imagen original B.

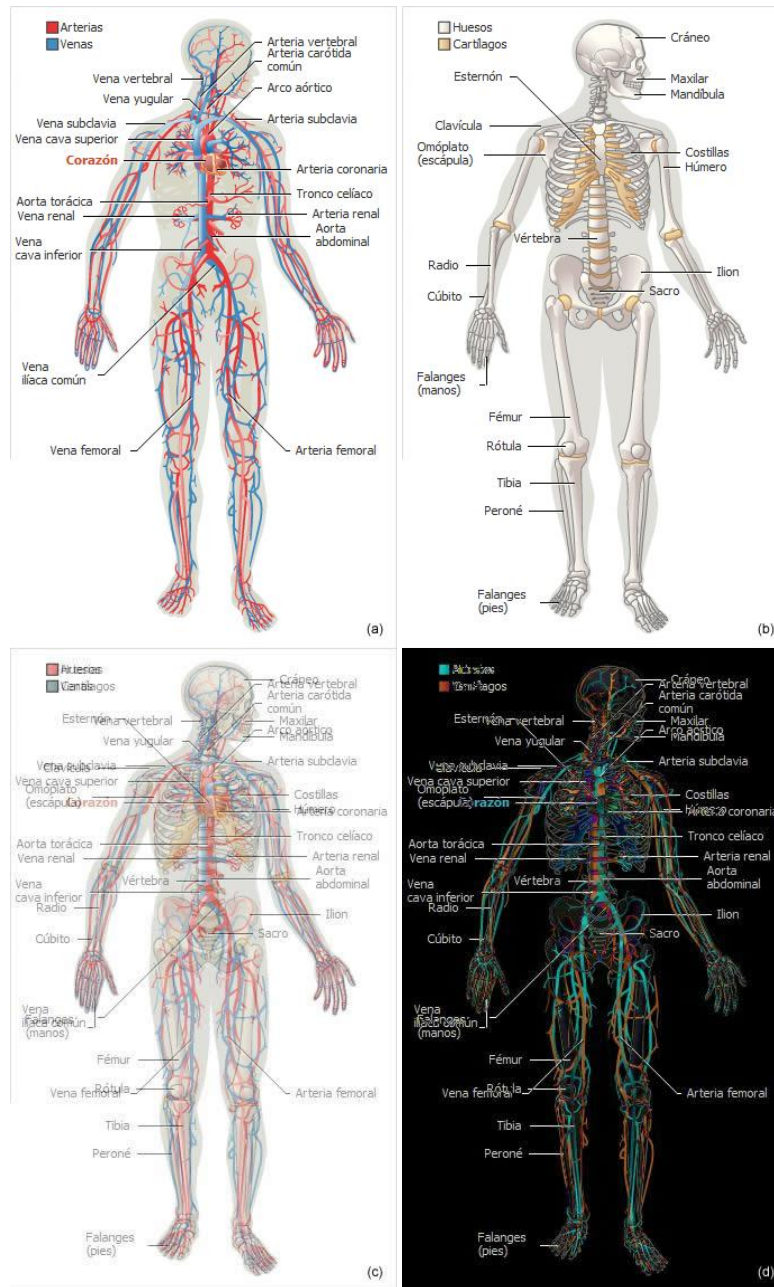


Figura 26. (a) Imagen original A. (b) Imagen original B. (c) Suma (A + B). (d) Resta (A - B).
(Anatomía humana, Enciclopedia Encarta)

II.4.4.2 Operaciones lógicas

Dadas dos imágenes es posible realizar sobre ellas operaciones lógicas tales como AND y OR, la cuales están dadas por las siguientes ecuaciones:

$$C_{x,y} = A_{x,y} \text{ AND } B_{x,y} \quad \text{AND lógico} \quad (\text{Ecuación 22})$$

$$C_{x,y} = A_{x,y} \text{ OR } B_{x,y} \quad \text{OR lógico} \quad (\text{Ecuación 23})$$

En la **Figura 27** se muestra el resultado de aplicar el AND lógico y el OR lógico entre la imagen original A y la imagen original B.

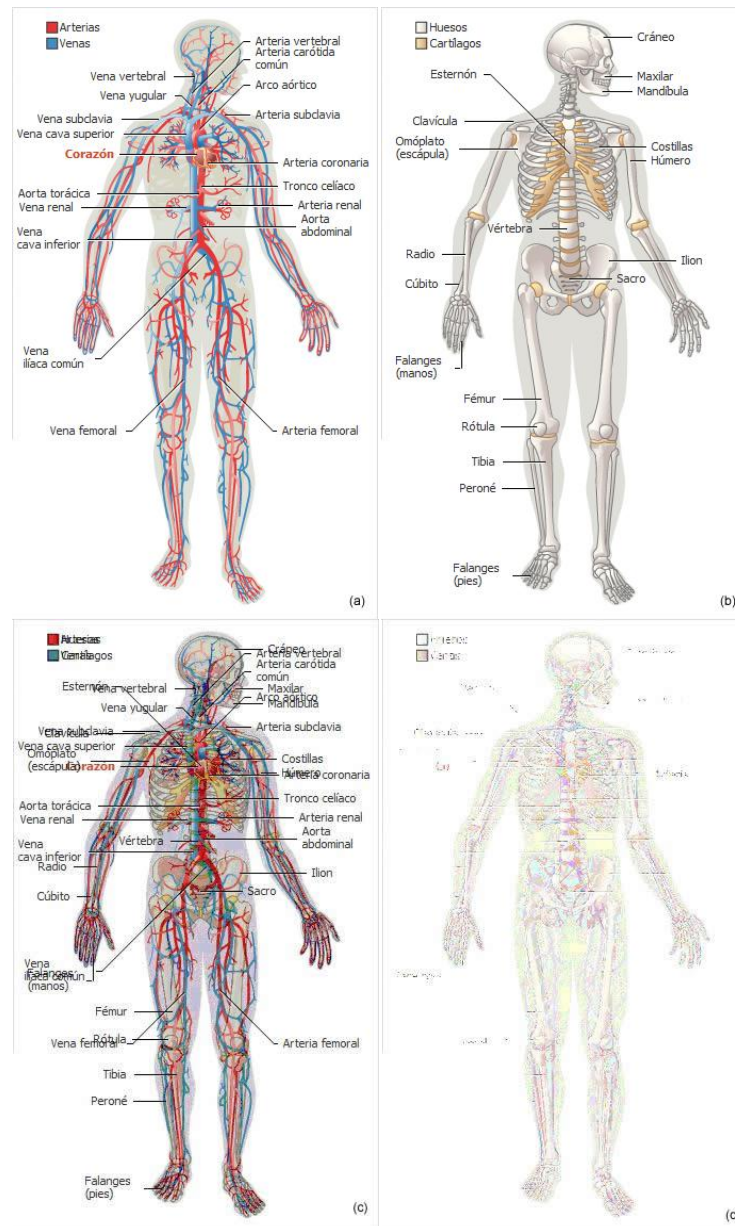


Figura 27. (a) Imagen original A. (b) Imagen original B. (c) AND lógico. (d) OR lógico. (Anatomía humana, Enciclopedia Encarta)

II.5 SOLUCIONES DE ALTO RENDIMIENTO PARA PROCESAMIENTO DIGITAL DE IMÁGENES

Existen distintas plataformas para la implementación de soluciones de alto rendimiento para procesamiento digital de imágenes, esto sin tomar en cuenta la computadora personal, entre estas se encuentran los FPGAs y los DSPs, a continuación se muestra un panorama de estas tecnologías y sus aplicaciones.

II.5.1 FPGA (Field Programmable Gate Array)

Los arreglos de bloques lógicos programables (FPGAs) son dispositivos semiconductores programables que están basados en una matriz de bloques lógicos configurables (CLB por sus siglas en inglés) conectados a través de interconexiones programables. De forma distinta a los circuitos integrados de aplicación específica (ASICs por sus siglas en inglés) donde el dispositivo está construido para un diseño en particular, los FPGAs pueden ser programados para cubrir distintos requerimientos de funcionalidad. (Xilinx, Inc., 2008a)

Aunque están disponibles FPGAs que son programables una sola vez, el tipo de FPGAs dominantes son los basados en memoria estática de acceso aleatorio (SRAM por sus siglas en inglés), los cuales pueden ser reprogramados mientras el diseño evoluciona.

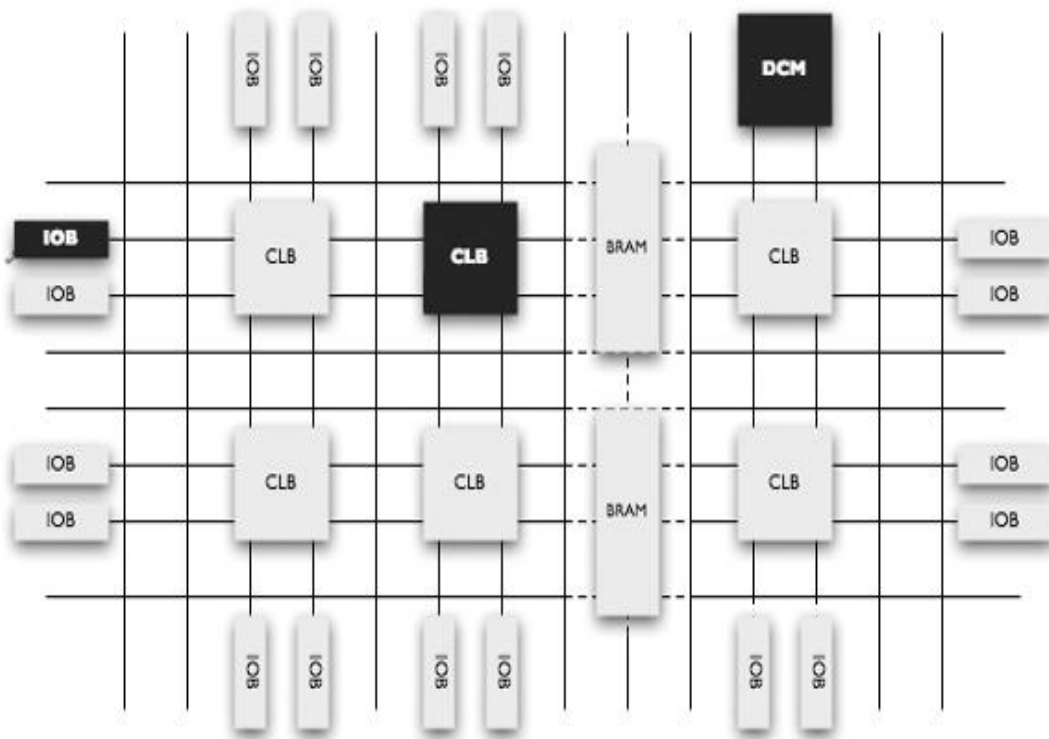


Figura 28. Estructura de un bloque de FPGA. (Xilinx, Inc., 2008a)

II.5.1.1 Características comunes de los FPGAs

Los FPGAs de hoy en día han evolucionado más allá de las capacidades básicas presentadas en sus predecesores, han incorporado bloques de uso común y funcionalidad como memoria de acceso aleatorio, administración de reloj, y procesadores digitales de señales. A continuación se muestran los componentes básicos en un FPGA.

II.5.1.1.1 Detalles del CLB (Configurable Logic Block)

El bloque lógico programable es la unidad lógica básica en un FPGA. El número exacto y las características varían de dispositivo a dispositivo, pero cada CLB consiste de una matriz de

interruptores configurable con cuatro o seis entradas, algunos circuitos de selección, y algunos flip-flops. La matriz de interruptores es altamente flexible y puede ser configurada para manejar lógica combinacional, registros de cambio o memoria de acceso aleatorio (RAM por sus siglas en ingles). El esquema de un CLB de alto nivel es mostrado en la **Figura 29**.

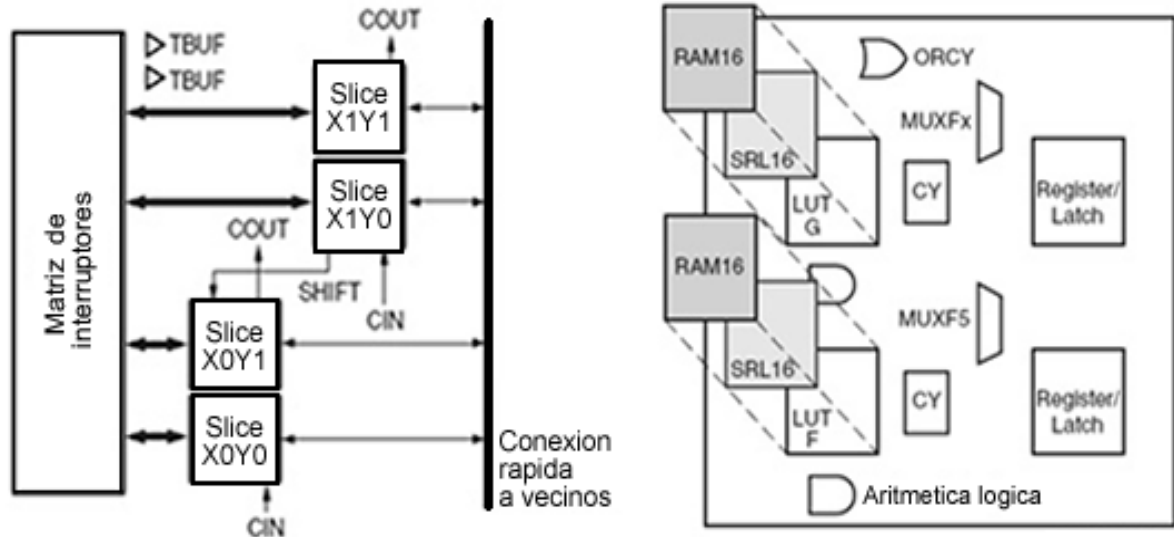


Figura 29. Esquema de un CLB de alto nivel. (Xilinx, Inc., 2008a)

II.5.1.1.2 Interconexión

Mientras que el CLB proporciona la capacidad lógica, la interconexión realiza el enrutamiento de las señales entre el CLB y las entradas/salidas, el software de diseño lleva a cabo la tarea de enrutamiento de interconexión ocultando esta tarea al usuario, a excepción de que se le indique lo contrario, esto reduce significativamente la complejidad en el diseño.

II.5.1.1.3 Detalles del IOB (Input Output Bank)

Los FPGAs de hoy en día proporcionan soporte para docenas de estándares de entrada/salida, así permite flexibilidad en soporte de entrada/salida, las entradas/salidas en los FPGAs están agrupadas en bancos, con cada banco se habilitan independientemente distintos estándares de entrada/salida.

II.5.1.1.4 Memoria

La mayoría de los FPGAs incluyen un bloque de memoria de acceso aleatorio (RAM), lo cual posibilita incluir el uso de memoria en el diseño. Los FPGAs de Xilinx proporcionan hasta 10 Mbits de memoria integrada en bloques de 36 Kbits los cuales soportan operaciones de doble puerto.

II.5.1.1.5 Detalles del DCM (Digital Clock Management)

La administración del reloj es proporcionada por la mayoría de los FPGAs en la industria (todos los FPGAs de Xilinx tienen esta característica).

II.5.1.2 Soluciones y aplicaciones con FPGAs.

Debido a su naturaleza programable, los FPGAs son apropiados para muchas aplicaciones entre las que destacan:

- *Aeroespacial y defensa*, FPGAs tolerantes a la radiación.
- *Automotriz*, soluciones para asistencia de manejo, confort, servicios de información y entretenimiento.
- *Industria/Ciencia/Medicina*, soluciones que apuntan a las necesidades específicas y nuevos retos en la automatización industrial, control de motores, y procesamiento de imágenes medicas.
- *Almacenamiento y servidores*, soluciones de procesamiento de datos para servidores, aplicaciones para almacenamiento, entre otros.
- *Comunicación inalámbrica*, RF, conectividad, soluciones de transporte y conectividad para equipo inalámbrico, estándares de direccionamiento tales como WCDMA, HSDPA, WiMAX, entre otros.

II.5.2 DSP (Digital Signal Processor)

El procesamiento digital de señales se ha convertido en el corazón de la revolución digital. Seguramente encontrara procesadores digitales de señales (DSP por sus siglas en ingles) en el corazón de los teléfonos celulares, reproductores de audio, reproductores de video, cámaras digitales, infraestructura telefónica, sistemas de control de motores, e incluso en equipo de seguridad biométrica. (Texas Instruments, 2008)

Cada una de estas aplicaciones trabaja con una señal o conjuntos de datos. Algunas décadas atrás, los desarrolladores descubrieron que procesar estas señales en el dominio digital ofrecía ventajas significativas en comparación con las señales analógicas, entre las cuales destacan las siguientes:

- Las señales digitales pueden ser enviadas largas distancias sin pérdida de datos.
- Trabajar con señales en el dominio digital simplifica el procesamiento de la señal, incluyendo efectos, ajuste de resolución, e intercambio entre formatos para incrementar la interoperabilidad entre dispositivos.
- Se pueden realizar copias tan buenas como la original de las señales digitales, tal como contenido de video o audio, mientras que el contenido de una copia analoga se degrada entre cada proceso de copiado.
- Los DSPs son programables, permitiendo a los desarrolladores continuamente mejorar la calidad sin rediseñar el hardware.

Los DSPs están optimizados para aplicaciones de procesamiento de señales comparado con procesadores de propósito general. Los DSPs ofrecen muchas características en su arquitectura que reduce el número de instrucciones necesarias para el eficiente procesamiento de una señal.

Los DSPs hacen posible muchas nuevas aplicaciones, consideremos cámaras de seguridad que pueden identificar y seguir distintos objetos, suministradores de energía que proporcionan energía con 0.5% de tolerancia, sistemas biométricos para reconocimiento de usuarios con niveles de seguridad requeridos en aplicaciones militares, mezcladores de audio profesionales que pueden procesar audio de alta calidad (192 Kbps) en tiempo real, y videoteléfonos que codifican y decodifican múltiples canales de audio y video.

II.6 FABRICANTES DE FPGAS

En los últimos años, en el mercado de los FPGAs se han colocado diversos fabricantes, cada uno se distingue por ofrecer dispositivos de capacidades únicas.

II.6.1 Xilinx

Xilinx, fundada en 1984 por Ross Freeman, Bernie Vonderschmitt, y Jim Barnett. Su misión y filosofía era crear un nuevo tipo de compañía, el concepto para un nuevo tipo de producto. (Xilinx, Inc., 2008b)

La tecnología que Xilinx impulso al principio fue considerada un concepto inesperado en 1984, inventado por el cofundador de Xilinx Roos Freeman, el nuevo semiconductor ahora conocido como FPGA, fue completamente una nueva forma de lógica programable. En 1985, Xilinx comercializa su primer producto, el XC2064, el cual fue el primer FPGA en la historia.

II.6.2 Altera

Altera, fundada en 1983 por Robert Hartmann, Michael Magranet, Paul Newhagen, y Jim Sansbury, los cuales fueron los pioneros de los PLDs (Programmable Logic Device), el EP300 fue el primer PLD en la historia, el cual abrió un nuevo segmento en el mercado de los semiconductores (Altera, 2008).

Hoy en día Altera ofrece FPGAs, CPLDs, y ASICs estructurados en combinación con herramientas de desarrollo.

II.6.3 Lattice Semiconductor

Lattice Semiconductor, Incorporada en Oregon, en 1983, reincorporada en Delaware en 1985.

En Lattice Semiconductor se diseñan, desarrollan y se comercializan semiconductores como los PLDs, microprocesadores como los ASICs, y arreglos de compuertas como los FPGAs. (Lattice Semiconductor Corporation, 2008)

II.6.4 QuickLogic

QuickLogic, fundada en 1988, es la compañía inventora y principal proveedora de plataformas flexibles de semiconductores para el desarrollo de dispositivos móviles. Utilizando una tecnología patentada de ultra bajo consumo de energía, QuickLogic proporciona soluciones para el mercado de dispositivos móviles. (QuickLogic, 2008)

II.6.5 Atmel

Atmel, fundada en 1984, es un líder mundial en diseño, manufactura y comercialización de semiconductores avanzados, incluyendo lógica avanzada, memoria no volátil, y circuitos integrados de radio frecuencia. (Atmel, 2008)

II.6.6 Achronix Semiconductor

Achronix Semiconductor, fundada en 2004, en la ciudad de Nueva York, reincorporo sus oficinas en San José, California en 2006. Achronix construye el FPGA más rápido, el cual usa una tecnología de circuitos patentada, la cual proporciona 2GHz de rendimiento. (Achronix Semiconductor Corporation, 2008)

II.6.7 MathStar

MathStar, fundada en Abril de 1997 por el veterano de la industria de comunicaciones Douglas M. Phil, con sus oficinas centrales en Portland, Oregon.

MathStar diseña, manufactura y comercializa una nueva clase de chips lógico programables llamados FPOAs (Field Programmable Objects Arrays), los cuales ofrecen alto rendimiento, circuitos integrados reprogramables y velocidad de reloj de hasta 1 GHz (MathStar, 2008)

II.7 FPGAS DE XILINX

Xilinx cuenta con dos tipos de FPGAs distintos, la serie Virtex y la serie Spartan las cuales a su vez se dividen en familias, a continuación se detallan cada una de ellas:

II.7.1 Serie Virtex.

La serie de FPGAs Virtex de Xilinx fue la primera línea en ofrecer un millón de compuertas. Fue introducido en 1998, la línea de producción de los FPGAs Virtex consiste en ofrecer la combinación de rendimiento, capacidad, e integración al menor costo posible. (Xilinx, Inc., 2008c)

II.7.1.1 Familia de FPGAs Virtex 5

Los FPGAs VirtexTM-5 ofrecen capacidades integradas que reducen los ciclos de diseño y reducen los costos del sistema.

- Hasta 330,000 celdas lógicas.
- Hasta 1,200 pines de entrada/salida.
- Emisores/Receptores seriales RocketIOTM GTP de bajo consumo de energía.
- Procesador PowerPC®440 para procesamiento integrado.
- Emisores/Receptores seriales RocketIOTM GTX de alto rendimiento.
- Puerto PCI Express y MAC Ethernet integrado.

Los FPGAs Virtex-5 ofrecen cuatro plataformas optimizadas.

- Virtex-5 LX, lógica de alto rendimiento.

- Virtex-5 LXT, lógica de alto rendimiento con conectividad serial de bajo consumo de energía.
- Virtex-5 SXT, para aplicaciones de procesamiento digital de señales y de acceso a memoria intensivo, incluye conectividad serial de alta velocidad.
- Virtex-5 FXT, para aplicaciones de procesamiento integrado y de acceso a memoria intensivo, incluye conectividad serial de bajo consumo de energía.

II.7.1.2 Familia de FPGAs Virtex 4.

Los FPGAs Virtex™-4 incorporan hasta 200,000 celdas lógicas, rendimiento de 500 MHz, procesador PowerPC®405 para procesamiento integrado, MAC Ethernet integrado, emisores/receptores seriales RocketIO™ multi-gigabit, y características del sistema capaces de entregar el doble de densidad, el doble de rendimiento, y la mitad de consumo de energía comparado con los FPGAs de generaciones anteriores.

Los FPGAs Virtex-4 ofrecen tres plataformas distintas, con un total de diecisiete dispositivos adaptados a los requerimientos de diferentes aplicaciones.

- Virtex-4 LX, lógica de alto rendimiento.
- Virtex-4 SX, para aplicaciones de procesamiento digital de señales y de acceso a memoria intensivo.
- Virtex-4 FX, para aplicaciones de procesamiento integrado, incluye conectividad serial de alta velocidad.

II.7.1.3 Familia de FPGAs Virtex II Pro

La familia de FPGAs Virtex-II Pro, se compone de dos FPGAs, el Virtex-II Pro y el Virtex-II Pro X.

Los FPGAs Virtex™-II Pro están basados en la tecnología de los FPGAs Virtex-II, proporcionan hasta dos procesadores PowerPC®405, incluye emisores/receptores seriales RocketIO™ multi-gigabit los cuales proporcionan conectividad de alta velocidad.

Xilinx ya no ofrece los FPGAs Virtex-II Pro X, por lo cual se recomienda no tomar en cuenta este tipo de FPGAs para nuevos diseños.

II.7.1.4 Familia de FPGAs Virtex II

Los FPGAs Virtex-II están desarrollados para diseños de alto rendimiento, la familia proporciona soluciones completas para telecomunicaciones, redes inalámbricas, video, y aplicaciones de procesamiento digital de señales. La familia de FPGAs Virtex-II se compone de once miembros los cuales se encuentran en un rango que va desde 40,000 hasta 8,000,000 de compuertas.

II.7.1.5 Familia de FPGAs Virtex-E y Virtex-EM

La familia de FPGAs VirtexTM-E fue introducida al mercado en 1998 y fabricada en 2.5V, la familia Virtex asigna el estándar para rendimiento y densidad en FPGAs. En 1999, la familia Virtex-E de 1.8V sale al mercado.

La familia de FPGAs Virtex-EM expande la capacidad de memoria para direccionar aplicaciones demandantes de este recurso tales como la manipulación de video de alta definición.

II.7.2 Serie Spartan

La serie de FPGAs Spartan de Xilinx son ideales para aplicaciones de bajo costo y de alto volumen. Con más de 120 millones de unidades vendidas desde 1998, La serie Spartan proporcionan una ruta fiable, eficiente, y de bajo costo para realizar de forma exitosa los requerimientos del sistema. (Xilinx, Inc., 2008d)

II.7.2.1 Spartan-3A DSP

La plataforma SpartanTM-3A DSP es ideal para la implementación de algoritmos de procesamiento digital de señales y aplicaciones de coprocesamiento que requieran un alto rendimiento en procesamiento digital de señales. La plataforma consiste de dos dispositivos:

- XC3SD3400A, proporciona hasta 30 GMAC/s (30 billones de operaciones de multiplicación acumulada) y hasta 2,200 Mbps de ancho de banda de la memoria.
- XC3SD1800A, proporciona hasta 20 GMAC/s (20 billones de operaciones de multiplicación acumulada).

II.7.2.2 Spartan-3AN

La plataforma no volátil SpartanTM-3AN proporciona lo mejor de dos mundos, características extensas y alto rendimiento proveniente de los FPGAs basados en SRAM, y la seguridad, ahorro en espacio y una sencilla configuración proveniente de los FPGAs no volátiles. Las principales características de la plataforma Spartan-3AN incluyen:

- FPGA de 90nm no volátil con numero serial electrónico, lo cual proporciona seguridad flexible y de bajo costo.
- Memoria Flash integrada con capacidad de hasta 11Mb.
- Administración de energía flexible, con hasta 40% de ahorro en modo suspendido.

II.7.2.3 Spartan-3A

La plataforma SpartanTM-3A fue diseñada desde cero con el propósito de servir como la punta de lanza en la industria para soluciones de conectividad. Esta plataforma soporta un amplio rango de estándares populares y emergentes de entrada/salida, junto con un extenso arreglo de soluciones prediseñadas.

II.7.2.4 Spartan-3E

La plataforma SpartanTM-3E es ideal para diseños programable lógicos en video digital, soluciones para la industria, soluciones médicas, soluciones en comunicación. La plataforma Spartan-3E incluye distintos FPGAs que van de 100,000 a 1,600,000 compuertas.

II.7.2.5 Spartan-3

La plataforma SpartanTM-3 es ideal para diseños donde se requiera alta densidad y un alto número de entradas/salidas, como en las aplicaciones altamente integradas de procesamiento de datos. La plataforma Spartan-3 incluye distintos FPGAs que van de 50,000 hasta 5,000,000 de compuertas.

II.7.2.6 Spartan-IIE

La plataforma SpartanTM-IIE proporciona todo lo necesario para crear productos optimizados en costo y flexibles. Con la plataforma Spartan-IIE se obtienen densidades de 50,000 hasta 600,000 compuertas.

II.7.2.7 Spartan-II

La plataforma SpartanTM-II proporciona 100,000 compuertas a velocidades de 200 MHz, proporcionando una flexibilidad en el diseño difícil de superar. Además la plataforma Spartan-II integra memoria RAM en el mismo chip y tecnología versátil de entrada/salida la cual soporta más de 16 estándares de alto rendimiento.

II.7.2.8 Spartan/XL

La plataforma SpartanTM/XL es la primera de bajo costo en ofrecer memoria RAM integrada en el chip y un amplio soporte de núcleos predefinidos. La plataforma Spartan/XL está disponible en densidades que van de 5,000 a 40,000 compuertas.

Capítulo III

III. IMPLEMENTACIÓN

III.1 INTRODUCCION

En este capítulo se detallan los distintos elementos utilizados para la implementación de este trabajo de tesis. Se hace una descripción del hardware utilizado, de las herramientas de desarrollo, y de los procesos llevados a cabo para la finalización de este trabajo de tesis.

III.2 HARDWARE

Para la implementación de este trabajo de tesis se utilizó la plataforma de evaluación ML403 de Xilinx, la cual permite investigar y experimentar con las características de los FPGAs VirtexTM-4 FX, con los cuales es posible implementar todos los componentes de un sistema con procesador embebido, a continuación se detallan sus principales características.

III.2.1 Plataforma de evaluación ML403

Las características más importantes de la plataforma de evaluación ML403 son las siguientes:

- FPGA Virtex-4 FX
- 64 MB DDR SDRAM, interface de 32 bit con velocidad de hasta 266 MHz
- Un reloj diferencial de entrada y un reloj diferencial de salida con conectores SMA
- Un reloj oscilador de 100 MHz
- LEDs y botones push
- Codificador de audio estéreo AC97, con línea de entrada, línea de salida, audífonos de 50 mW y entrada de micrófono
- Puerto serial RS-232
- Pantalla LCD de 16 caracteres con 2 líneas
- Memoria de 4 Kb IIC EEPROM
- Salida VGA
- Conector PS/2 para ratón y teclado
- SystemACETM CompactFlash con conector tipo CompactFlash I/II
- SRAM sincronizada de 8 Mb con bus de datos de 32 bit, sin bit de paridad.
- Memoria Flash Intel StrataFlash de 8 Mb
- Emisor-Transmisor Ethernet 10/100/1000

- Interface USB
- Puerto de configuración para uso con cable paralelo III o cable paralelo IV
- Proveedores de energía para todos los voltajes necesarios
- Adaptador de 5V@3A AC
- LED indicador de encendido

La **Figura 30** muestra el diagrama de bloques de la plataforma de evaluación ML403.

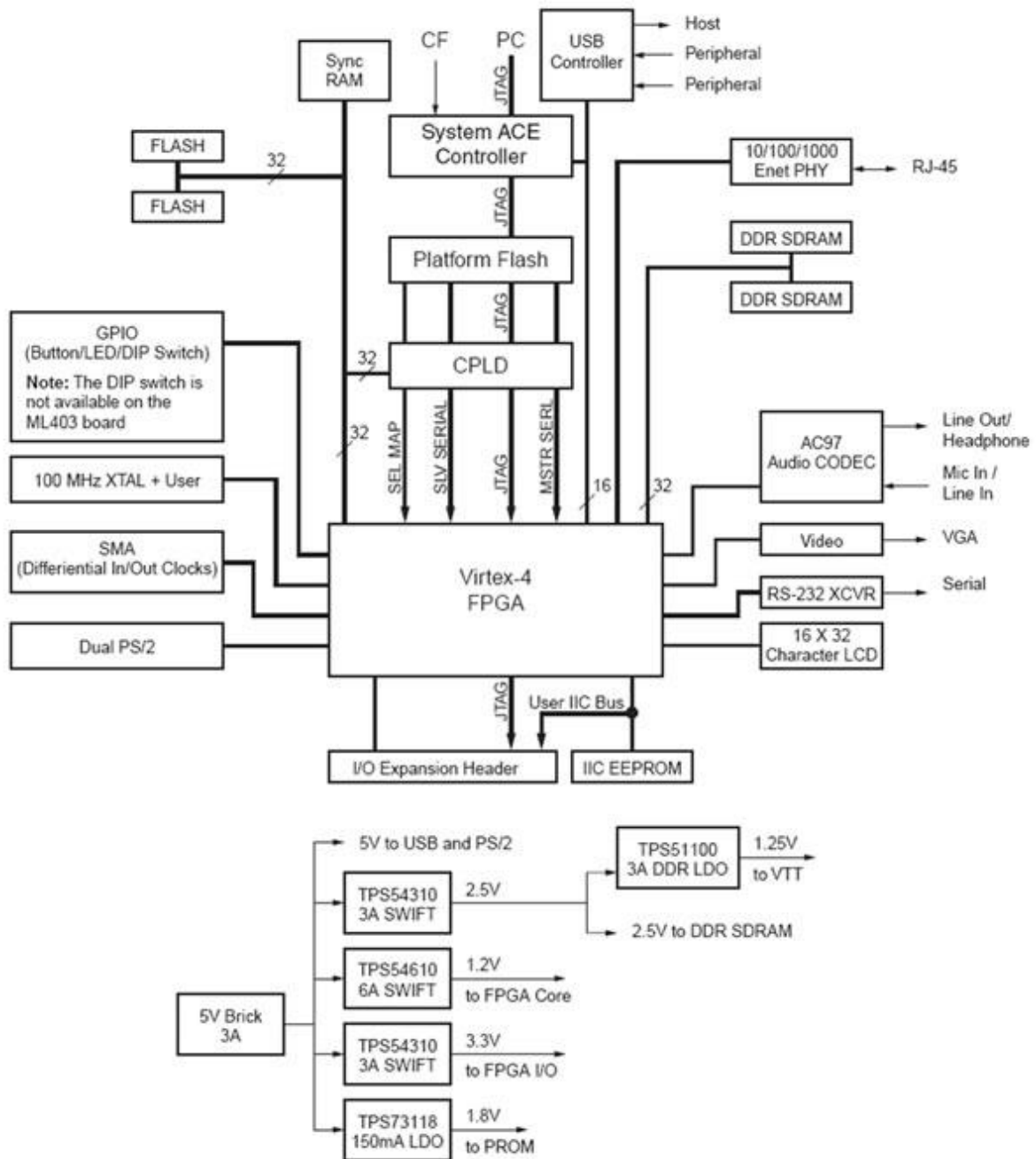


Figura 30. Diagrama de bloques de la plataforma de evaluación ML403. (Xilinx, Inc., 2006)

La **Figura 31** y **Figura 32** muestran la plataforma de desarrollo ML403 (parte frontal y trasera respectivamente), más adelante se detallan cada una de los componentes numerados en las figuras.

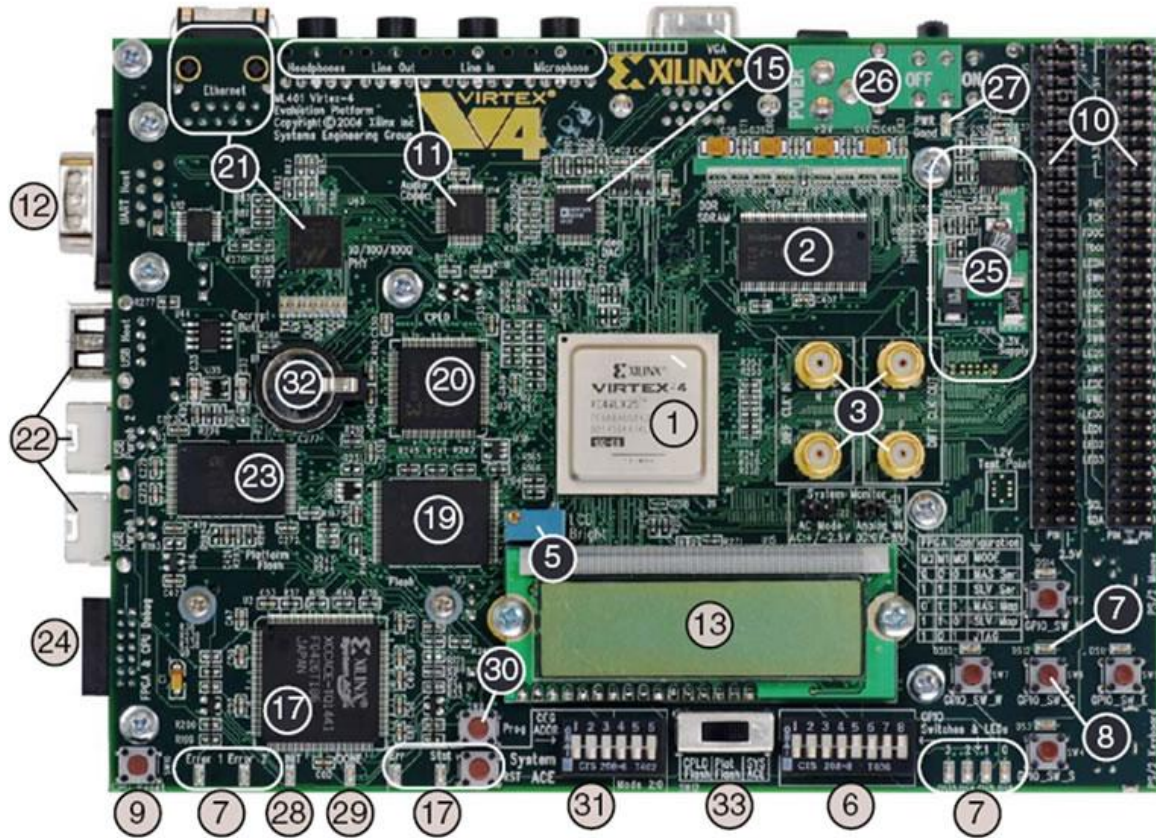


Figura 31. Descripción detallada de los componentes de la plataforma de evaluación ML403 (Parte frontal). (Xilinx, Inc., 2006)

III.2.1.1 Componentes de la plataforma de evaluación ML403

1. FPGA Virtex-4 FX

Un FPGA Virtex-4 FX (XC4VFX12-FF668-10) está instalado en la plataforma de evaluación ML403.

2. DDR SDRAM

La plataforma de evaluación contiene 64 MB de DDR SDRAM utilizando dos chips *Infineon* HYB25D256160BT-7 (U4 y U5). La tarjeta puede soportar hasta 256 Mb de memoria DDR SDRAM total si chips más grandes son instalados. Un pin de dirección extra está presente en la tarjeta para soportar hasta 1 Gb de memoria DDR.

3. Reloj diferencial de entrada y reloj diferencial de salida

Señales de reloj de alta precisión pueden ser introducidas al FPGA utilizando señales de reloj diferenciales traídas a través de los conectores SMA de 50Ω.

4. Sockets osciladores

La plataforma de evaluación contiene dos sockets osciladores de cristal.

5. Ajuste de contraste y brillo para el LCD

El potenciómetro R1 incluido en la plataforma de evaluación ajusta el contraste de la imagen de los caracteres del LCD.

6. Interruptores DIP (Activos en alto)

La plataforma de evaluación contiene ocho interruptores de propósito general que están conectados a las entradas y salidas de usuario del FPGA.

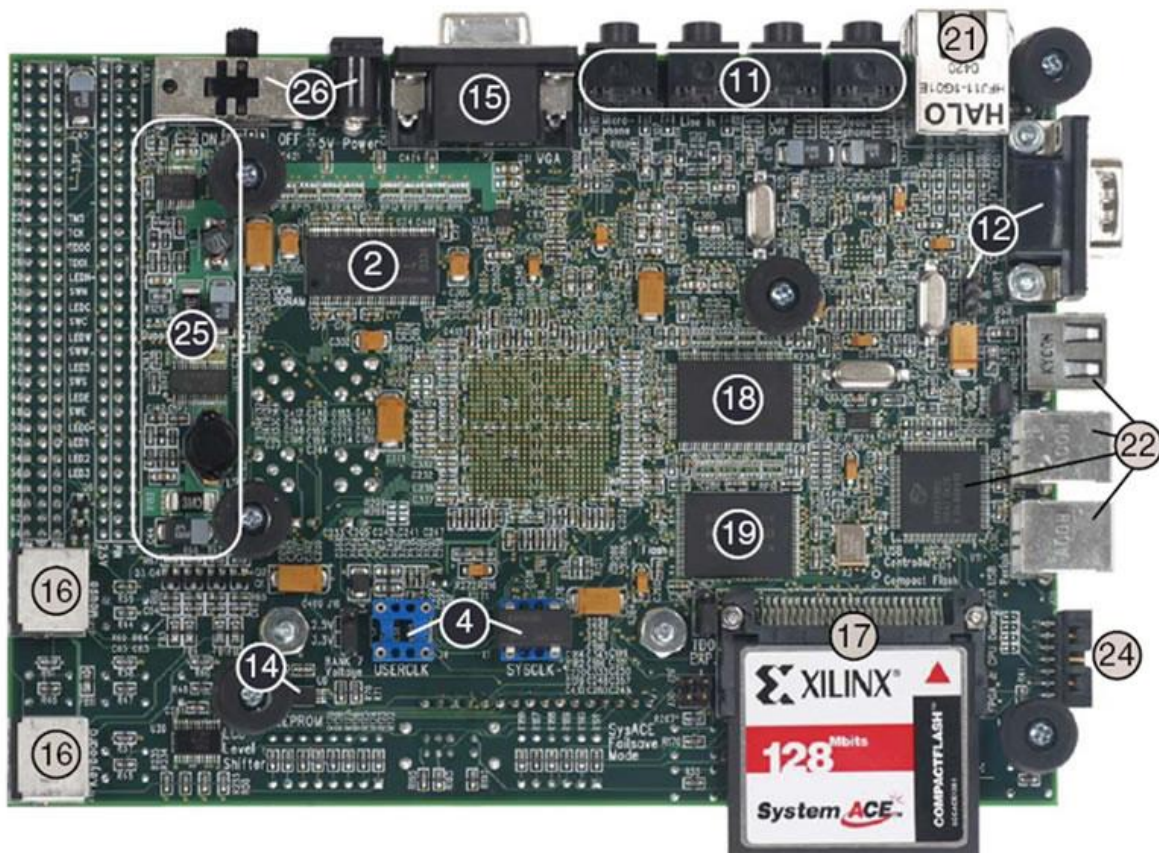


Figura 32. Descripción detallada de los componentes de la plataforma de evaluación ML403 (Parte trasera). (Xilinx, Inc., 2006)

7. LEDs de usuario y de error (*Activos en alto*)

La plataforma de evaluación contiene 10 LEDs, controlados directamente por el FPGA.

- Cuatro LEDs verdes de propósito general alineados en fila.
- Cinco LEDs verdes posicionados junto a los botones de presión Norte, Este, Sur, Oeste, Centro.
- Un LED rojo que es utilizado para señalar condiciones de error.

8. Botones de presión de usuario (*Activos en alto*)

La plataforma de evaluación contiene cinco botones de presión, los cuales están disponibles para propósito general y están ordenados en Norte, Este, Sur, Oeste, Centro.

9. Botón de reinicio del CPU (*Activo en bajo*)

La plataforma de evaluación incluye un botón de reinicio del CPU que se puede utilizar como botón de reinicio del sistema o de usuario.

10. Cabeceras de expansión

La plataforma de evaluación contiene cabeceras de expansión para la fácil extensión o adaptación de la tarjeta para otras aplicaciones.

11. Codificador de audio estéreo AC97

La plataforma de evaluación contiene un codificador de audio lo que permite el procesamiento de audio. El semiconductor codificador de audio de National LM4550 soporta audio estéreo de 16 bits con muestreo de hasta 48 KHz.

12. Puerto serial RS-232

La plataforma de evaluación contiene un puerto serial macho DB-9 RS-232, lo que permite al FPGA comunicarse de forma serial con otro dispositivo.

13. Pantalla LCD de 16 caracteres y 2 líneas

La plataforma de evaluación contiene una pantalla LCD (Lumex LCM-S01602DTR/M) para desplegar texto. La interface de datos al LCD está conectada al FPGA para soportar solo el modo de 4 bits.

14. Bus IIC con 4 Kb EEPROM

Una IIC EEPROM (Microchip Technology 24LC04B-I/ST) es proporcionada en la plataforma de evaluación para almacenar información no volátil como la dirección MAC del transmisor-receptor Ethernet.

15. Salida VGA

La salida VGA soporta un monitor externo, mediante un chip de video (Analog Devices ADV7125KST140) de 140 MHz y 15 bits de bus de datos conectados al FPGA.

16. Puertos PS/2 para ratón y teclado

La plataforma de evaluación contiene dos puertos PS/2 uno para ratón y otro para teclado.

17. Conector SystemACE y CompactFlash

El controlador de configuración Xilinx SystemACE CompactFlash permite una tarjeta CompactFlash tipo I o tipo II, para programar el FPGA a través del puerto JTAG.

18. SRAM síncrona ZBT

La memoria SRAM síncrona ZBT (Cypress CY7C1354B o compatible) proporciona memoria externa de alta velocidad al FPGA. La memoria está organizada como 256K x 36 bits.

19. Chips Flash lineales

Dos chips lineales de 32 Mb (Micron MT28F320J3RG-11 ET) están instalados en la plataforma de evaluación para un total de 8 Mb de memoria flash. Estos chips de memoria flash son compatibles con los chips Intel StrataFlash.

20. CPLD XC95144XL de Xilinx

El CPLD de Xilinx está conectado a la memoria flash y a las señales de configuración del FPGA. Esta conexión de CPLD soporta aplicaciones donde la memoria flash programa el FPGA.

21. Emisor-Transmisor Ethernet 10/100/1000

La plataforma de evaluación contiene un dispositivo Marvell Alaska PHY (88E1111) que opera a 10/100/1000 Mb/s. La tarjeta soporta los modos de interface MII, GMII, y RGMII con el FPGA.

22. Controlador USB con host y puertos para periféricos

La plataforma de evaluación contiene un controlador host USB el cual proporciona conectividad USB para la tarjeta. El controlador USB soporta los modos de operación de host y de periféricos

23. Plataforma flash XCF32P de Xilinx configurado como dispositivo de almacenamiento

La plataforma flash XCF32P de Xilinx ofrece una configuración conveniente y fácil de utilizar para el FPGA.

24. Puerto de configuración JTAG

El puerto de configuración JTAG de la plataforma de evaluación, permite la programación de dispositivos y la depuración del FPGA.

25. Proveedor de energía integrado

Los proveedores de energía integrados en la plataforma de evaluación generan 1.2V, 1.25V, 1.8V, 2.5V, y 3.3V para alimentar los distintos componentes en la tarjeta.

26. Adaptador AC y entrada de energía Interruptor/Jack

La plataforma de evaluación incluye un adaptador AC de 15W (5V@3A).

27. LED indicador de encendido

El LED PWR se ilumina cuando los proveedores de energía de 1.2V, 2.5V, y 3.3V están operando en condiciones óptimas.

28. LED de inicio

El LED INIT se ilumina al encendido de la plataforma de evaluación para indicar que el FPGA se inicio correctamente y se completo su proceso interno de encendido.

29. LED de terminado

El LED DONE indica el estado del pin DONE en el FPGA. Se ilumina cuando el FPGA es satisfactoriamente configurado.

30. Interruptor de programa

Se utiliza para borrar el FPGA.

31. Dirección de configuración e interruptores DIP

Los interruptores DIP de 6 posiciones controlan la dirección y el modo de configuración del FPGA.

32. Batería de la llave de encriptación

Una batería incluida en la plataforma de evaluación está conectada al pin V_{BATT} del FPGA para mantener la llave de encriptación del FPGA.

33. Interruptor selector de la fuente de configuración

El Interruptor selector de la fuente de configuración selecciona entre los distintos métodos de programación del FPGA que son SystemACE, Platform Flash, y Linear Flash / CPLD.

III.3 HERRAMIENTAS DE DESARROLLO

Para la implementación de este trabajo de tesis se utilizó *Xilinx Platform Studio (XPS)*, el cual proporciona un conjunto de herramientas de diseño basadas en una estructura común. XPS es el software de diseño proporcionado en el *Embedded Development Kit (EDK)* de Xilinx. XPS consiste de una interfaz y todas las herramientas necesarias para desarrollar los componentes de hardware y software de un sistema con procesador embebido implementado en un FPGA de Xilinx. A continuación se detallan algunas de las principales características del Xilinx EDK.

III.3.1 EDK de Xilinx

El EDK de Xilinx consta muchas herramientas, entre las cuales destacan por su mayor utilidad las siguientes:

III.3.1.1 XPS (*Xilinx Platform Studio*)

Proporciona un ambiente integrado para la creación de especificaciones de software y hardware para sistemas con procesadores embebidos en procesadores MicroBlaze™ y PowerPC®.

También proporciona un editor y una interfaz para la administración de proyectos, la creación y la edición de código fuente. XPS ofrece un sistema editor gráfico para la conexión de procesadores, periféricos y buses.

III.3.1.2 Herramienta para la generación de hardware (*Platgen*)

La herramienta para la generación de hardware (*Platgen*) personaliza y genera el sistema de procesador embebido, en forma de archivos de hardware (*netlist*).

Por defecto, *Platgen* sintetiza cada instancia de núcleo IP encontrada en el diseño de hardware embebido utilizando el compilador XST. *Platgen* también genera los archivos HDL (*Hardware Description Language*) a nivel de sistema, los cuales interconectan todos los núcleos IP.

III.3.1.3 Asistente para la creación de un sistema base (*BSB por sus siglas en inglés*)

El asistente para la creación de un sistema base permite crear rápidamente un diseño embebido funcional, utilizando cualquier característica de las plataformas de evaluación soportadas o utilizando funcionalidad básica común a la mayoría de los sistemas embebidos.

Después de crear un sistema básico, se puede personalizar utilizando las herramientas incluidas en el XPS.

III.3.1.4 Herramienta de generación de librerías

La herramienta de generación de librerías, realiza la configuración de las librerías de controladores de dispositivos, sistema de archivos, y los manejadores de interrupción para el sistema con procesador embebido, creando una plataforma de software.

III.3.1.5 Herramientas de desarrollo de software GNU

Aplicaciones de software embebidas escritas en C, C++, o ensamblador son compiladas utilizando la cadena de herramientas de compiladores GNU. La cadena de herramientas GNU es parte del EDK personalizado para los procesadores PowerPC y MicroBlaze.

III.3.1.6 SDK (Software Development Kit)

El SDK es una interfaz complementaria al XPS y proporciona un ambiente de desarrollo para proyectos de software. SDK está basado en el estándar abierto Eclipse.

III.4 DISEÑO DEL SISTEMA

Para la implementación de este trabajo de tesis se realizó el diseño de un sistema con procesador embebido. Utilizando la plataforma de evaluación ML403 de Xilinx y la herramienta de desarrollo EDK de Xilinx, la cual nos permite desarrollar los componentes de hardware y software del sistema con procesador embebido.

Con el diseño de un sistema con procesador embebido se pueden aprovechar al máximo los recursos del FPGA utilizando un enfoque Hardware-Software, con esto las tareas complejas y que requieran gran cantidad de recursos se pueden implementar en hardware y las tareas simples en software. El FPGA Virtex-4 FX incluye un microprocesador PowerPC junto con sus recursos estándar y establece una interfaz para su comunicación, esto reduce el tiempo de comunicación entre los métodos de Hardware-Software. El diseño de este trabajo de tesis se divide en las siguientes fases:

- Creación de la plataforma de hardware
- Creación de la plataforma de software
- Creación de la aplicación de software
- Verificación del software (depuración)
- Verificación de la plataforma de software y hardware (simulación)

III.4.1 Creación de la plataforma de hardware

La plataforma de hardware consiste de uno o más procesadores y periféricos conectados al bus del procesador. El archivo de especificación de hardware del microprocesador (MHS, por sus siglas en inglés) define la arquitectura del sistema, periféricos, y los procesadores embebidos. Además define la conectividad del sistema, la dirección de cada uno de los periféricos en el sistema y las opciones de configuración para cada periférico.

Para la creación de la plataforma de hardware se utilizó el procesador embebido PowerPC incluido en el FPGA Virtex-4 FX, controlador de memoria RAM, controlador de salida VGA, controlador de memoria Compact Flash y los respectivos buses para la interconexión de dichos periféricos. En la **Figura 33** se muestra un diagrama de bloques de los distintos periféricos utilizados en el diseño de hardware del sistema, además de la forma en la que se interconectan dichos periféricos junto con el procesador, el diagrama es generado por la herramienta XPS de Xilinx.

La **Figura 34** muestra la vista de montaje del sistema, la cual permite visualizar la plataforma de hardware del sistema ordenada de forma jerárquica y agrupada por la instancia del núcleo IP. El panel de conectividad, es una representación grafica del bus de conectividad del hardware del sistema. Cada línea vertical representa un bus, y cada línea horizontal representa la interface de bus para un núcleo IP.

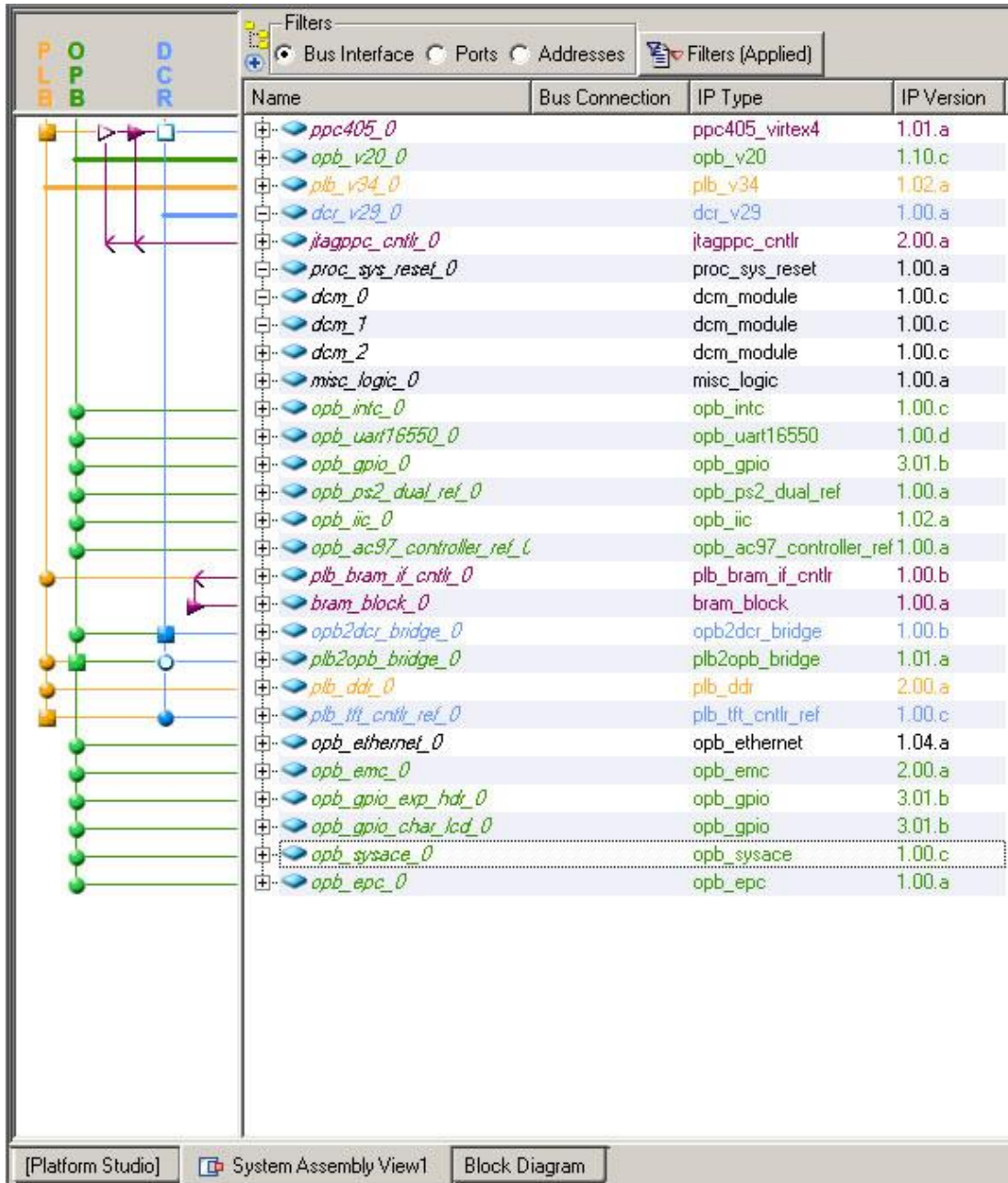


Figura 34. Detalle de los distintos periféricos, buses y procesador, mostrando la interconexión entre los distintos periféricos.

A continuación se detallan los núcleos IP más relevantes utilizados en la plataforma de hardware del sistema, los cuales aparecen en la **Figura 34**.

ppc405_virtex4, el bloque de procesador PowerPC 405 para dispositivos Virtex-4 FX contiene un núcleo de procesador PPC405F6, lógica de memoria en el chip, y un controlador APU. El envoltorio del PPC405 para Virtex-4 FX adapta la configuración de parámetros y señales de entrada-salida del bloque de procesador para compatibilidad con el ambiente de diseño del EDK.

plb_ddr, el controlador síncrono de DRAM (SDRAM) PLB DDR (Processor Local Bus Double Data Rate) de Xilinx es un núcleo IP para los FPGAs Virtex-II Pro y Virtex-4. Proporciona un controlador DDR SDRAM el cual se conecta al bus local del procesador (PLB, por sus siglas en inglés) y proporciona control de la interface para SDRAMs DDR.

plb_tft_cntlr_ref, el controlador PLB TFT LCD es un controlador de pantalla TFT o VGA con resolución de 640x480 píxeles. Es capaz de mostrar hasta 256 mil colores y está diseñado para pantallas TFT, pero también puede ser usado para pantallas VGA mediante el puerto de salida VGA de la plataforma de evaluación ML403. El diseño contiene una interfaz maestra PLB que lee los datos de video desde un dispositivo fijo de memoria y muestra la información en una pantalla TFT. El diseño también contiene una interface para configurar el controlador.

opb_gpio, el diseño OPB GPIO proporciona una interface de entrada-salida de propósito general. El OPB GPIO (On-Chip Peripheral Bus General Purpose Input Output) puede ser configurado como un dispositivo de canal simple o de canal doble. El ancho del canal es configurable y cuando ambos canales están habilitados, el ancho del canal es el mismo para ambos.

opb_sysace, el controlador de interfaz OPB System ACE es la interfaz entre el bus local del procesador (PLB, por sus siglas en inglés) y la interfaz del microprocesador (MPU, por sus siglas en inglés) del periférico System ACE Compact Flash.

III.4.2 Creación de la plataforma de software

La plataforma de software es una colección de controladores de software y, opcionalmente, el sistema operativo en el cual se va a construir la aplicación. Debido a la naturaleza fluida de la plataforma de hardware, se pueden crear una variedad de plataformas de software para cualquier plataforma de hardware dada.

Para la creación de la plataforma de software, el sistema operativo por defecto es Standalone, el cual proporciona los paquetes de soporte para las distintas tarjetas de desarrollo (Standalone BSP). Esto significa que no existe sistema operativo entre la aplicación y el hardware de la plataforma. La aplicación de software opera en la plataforma de hardware a través de llamadas directas a los controladores.

La aplicación de software puede utilizar los controladores de los dispositivos y algunas librerías básicas. Los controladores por defecto y los periféricos presentes en la plataforma de hardware son asignados al procesador.

El MicroKernel de Xilinx (XMK, por sus siglas en ingles), es la entidad que representa el sistema colectivo de software, lo cual comprende las librerías estándar de C, Kernel de Xilinx, Standalone BSP (Board Support Package), librería de Xilinx MFS (Memory File System), archivos de la librería de Xilinx, y controladores de la librería de Xilinx.

Existen tres entidades diferentes en el XMK, con las cuales las aplicaciones del usuario pueden interactuar con las librerías Estándar de C, librerías de matemáticas, librerías de Xilinx (libXil), y con los sistemas operativos Standalone o Kernel de Xilinx (XilKernel).

El soporte de la librería estándar de C consiste de la librería *libc*, la cual contiene las funciones estándar de C, *stdlib* y rutinas de cadenas. La librería *math* proporciona las rutinas de matemáticas estándar.

Las librerías de Xilinx consisten de:

- Librerías de Xilinx de controladores, controladores de los dispositivos de Xilinx.
- Librerías del sistema de archivos de memoria (LibXil MFS), Sistema de archivos de memoria de Xilinx.

Standalone BSP y XilKernel son dos opciones de sistemas operativos que son proporcionados en el XMK y que pueden ser incluidos en la plataforma de software.

La estructura del XMK es mostrada en la **Figura 35**, la aplicación del usuario puede interactuar con los componentes del XMK en distintas maneras. Las librerías son independientes entre sí, exceptuando algunas interacciones.

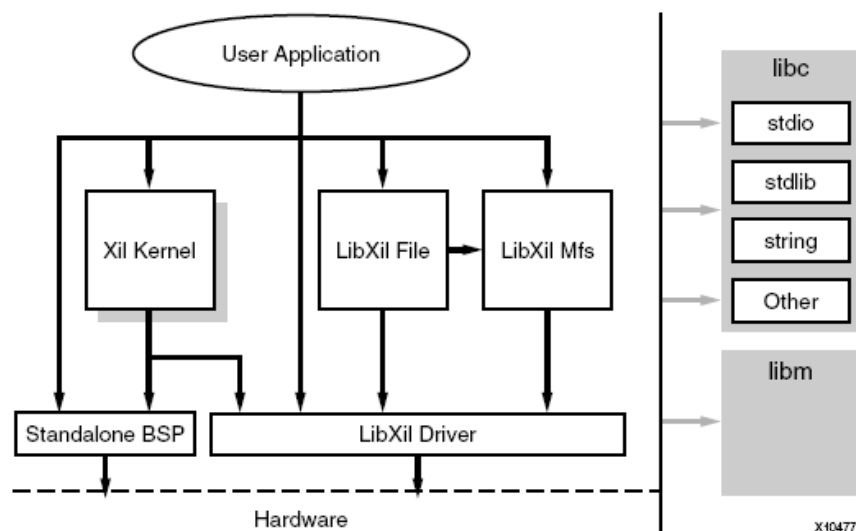


Figura 35. Estructura del MicroKernel de Xilinx (XMK).

La **Figura 36** muestra la configuración de la plataforma de software, se puede notar el uso de Standalone como sistema operativo en su versión 1.00.a, el cual proporciona funciones relacionadas con el procesador y funciones parecidas a las de un sistema operativo tales como entrada y salida estándar. En la **Figura 36** también se muestra la versión del procesador PowerPC utilizado.

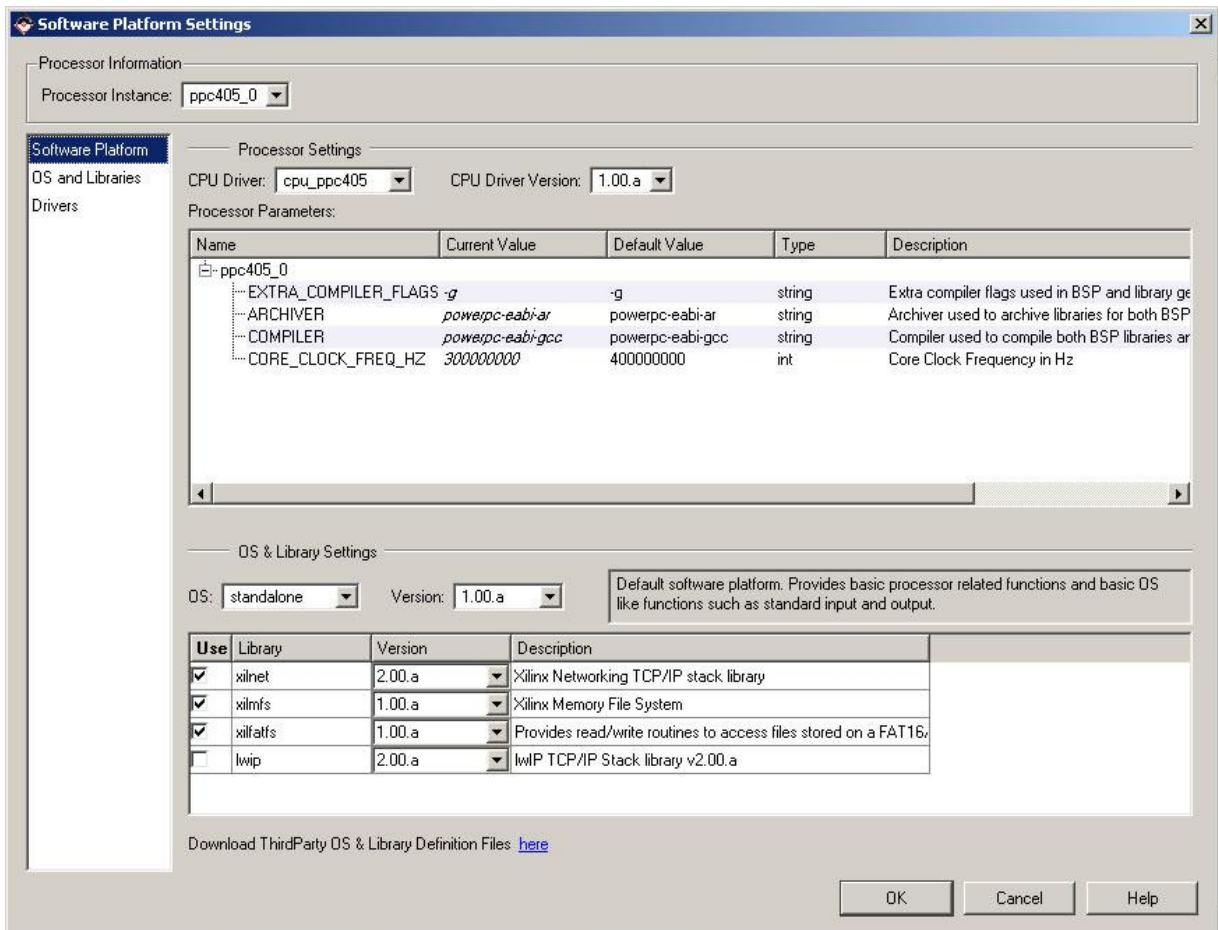


Figura 36. Configuración de la plataforma de software.

La **Figura 37** muestra la configuración básica de Standalone, sobresale la configuración de la salida estándar y la entrada estándar, el uso de excepciones, entre otras características. También muestra la configuración de las librerías entre las cuales destacan:

XilFATfs, librería de acceso al sistema de archivos la cual proporciona acceso de lectura/escritura a los archivos almacenados en dispositivos System ACE™ CompactFlash de Xilinx® o dispositivos IBM Microdrive.

XilMFS, librería de sistema de archivos de memoria, proporciona la capacidad al usuario de administrar la memoria del programa en forma de archivos manejables.

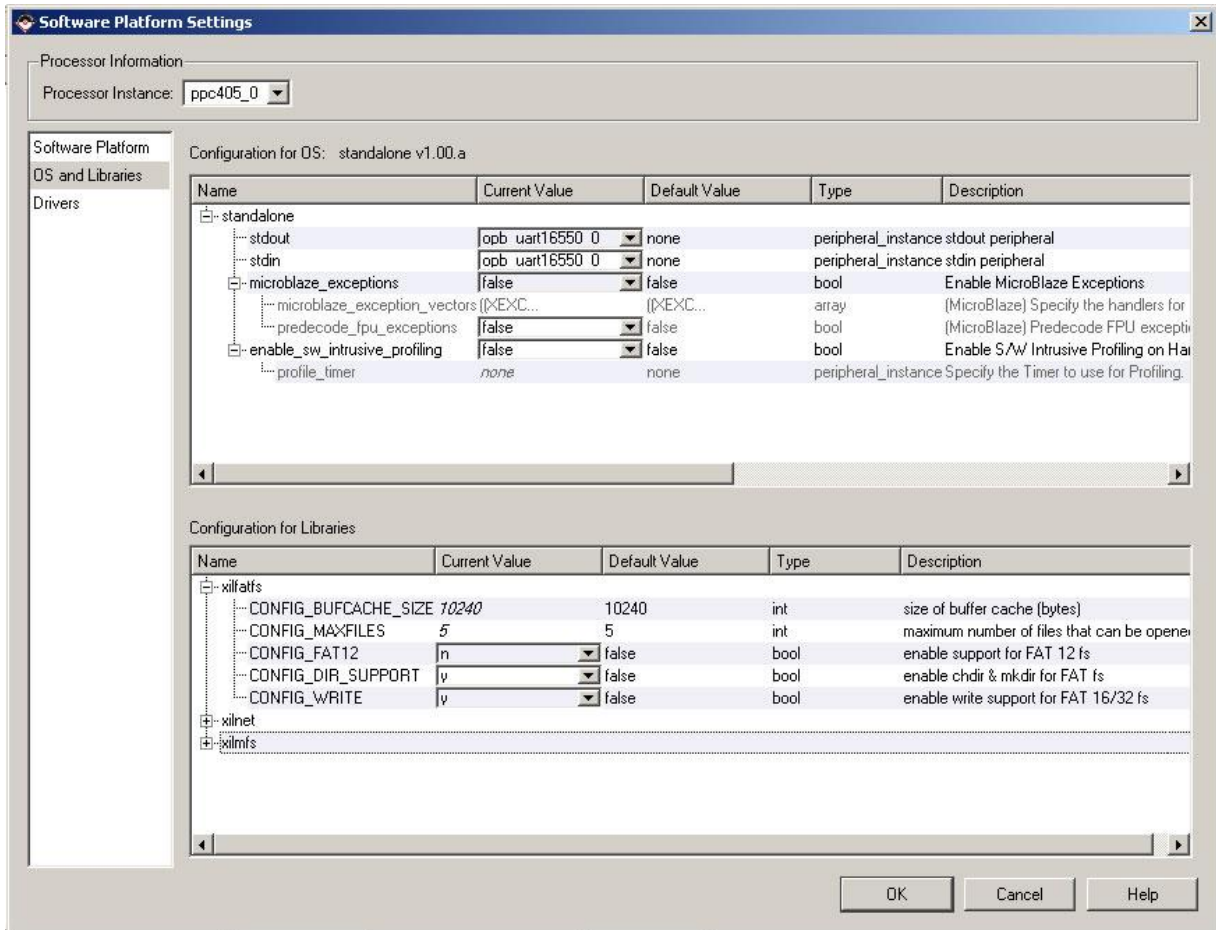


Figura 37. Configuración del Sistema Operativo y de las librerías.

La **Figura 38** muestra la asignación de controladores, para cada uno de los periféricos presentes en la plataforma de hardware.

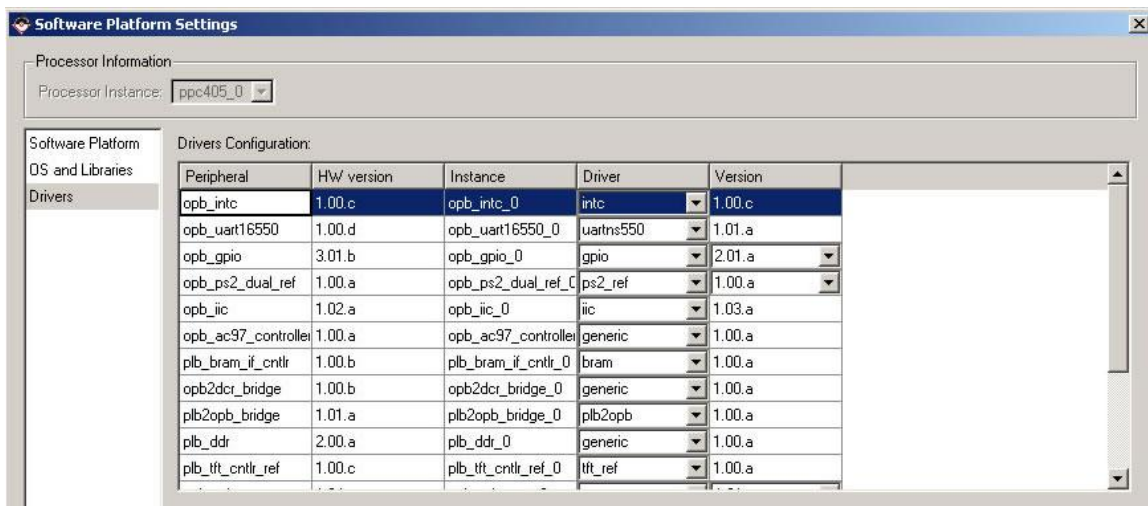


Figura 38. Controladores asignados a los distintos periféricos presentes en la plataforma de hardware.

III.4.3 Creación de la aplicación de software

Después de crear la plataforma de software, se debe generar la aplicación de software, la cual está constituida por código fuente escrito en C o C++, incluyendo archivos de código y archivos de cabecera.

Para la implementación de este proyecto, se desarrollaron los siguientes archivos de código escritos en lenguaje C:

- main.c, contiene la función principal y maneja la interacción del usuario con la aplicación.
- main_graficosbasicos.c, contiene la implementación de las funciones necesarias para generar gráficos básicos como rectángulo, líneas, píxeles, cadenas de caracteres en pantalla.
- main_graficos.c, contiene la implementación de las funciones necesarias para generar gráficos avanzados como botones y elementos de captura de parámetros.
- filtros_puntuales.c, contiene la implementación de los distintos algoritmos de operaciones individuales, analizados en el capítulo II.
- filtros_regionales.c, contiene la implementación de los distintos algoritmos de operaciones de vecindad, analizados en el capítulo II.
- operaciones_geometricas.c, contiene la implementación de los distintos algoritmos de operaciones geométricas entre imágenes, analizados en el capítulo II.
- imagen.c, contiene la implementación de las funciones necesarias para manipular imágenes de mapa de bits.
- util.c, contiene la implementación de las funciones necesarias para la manipulación de cadenas.

También se desarrollaron los siguientes archivos de cabecera, los cuales contienen las firmas de las funciones implementadas en los archivos de código fuente.

- main_menu.h
- main_graficosbasicos.h
- main_graficos.h
- imagen.h
- filtros_puntuales.h
- filtros_regionales.h
- operaciones_geometricas.h
- util.h

La **Figura 39** muestra los distintos archivos utilizados para la implementación de este proyecto.

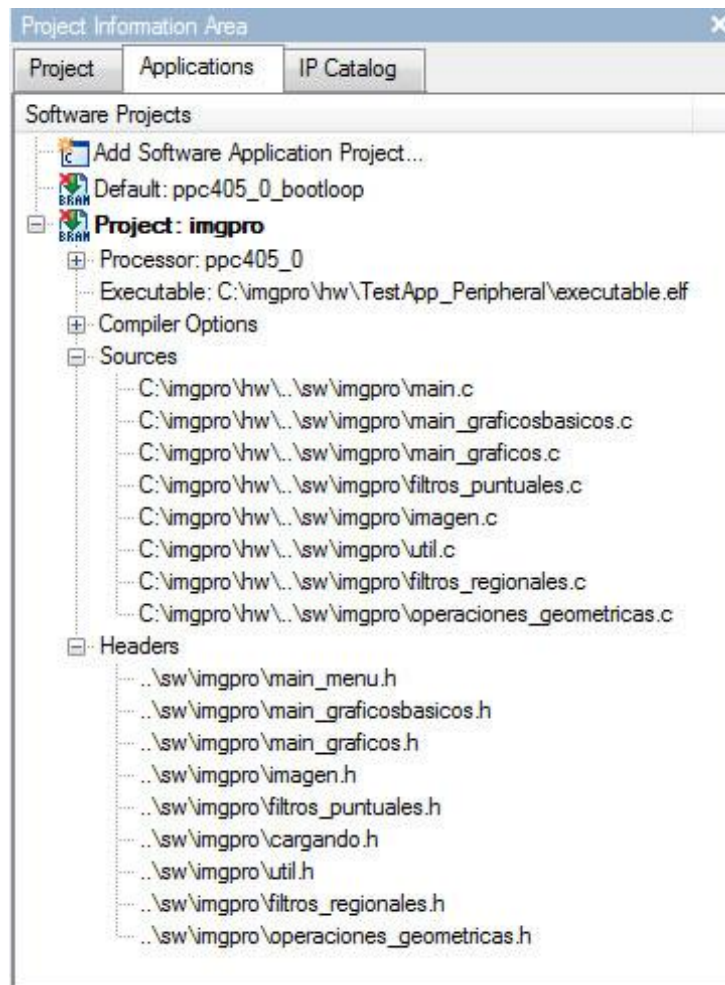


Figura 39. Archivos de la aplicación de software.

III.4.4 Verificación del software (depuración)

El EDK proporciona las herramientas del depurador GNU (GDB, por sus siglas en inglés) y las herramientas de depuración del SDK (Platform Studio Software Development Kit). Estas herramientas soportan las arquitecturas de los procesadores PowerPC 405, PowerPC 440, y MicroBlaze. Se pueden depurar programas ejecutados en la tarjeta de hardware, simulador de conjunto de instrucción de ciclo preciso, y en el sistema de plataforma virtual de ciclo preciso MicroBlaze.

Las herramientas de depuración utilizan el microprocesador depurador de Xilinx (XMD, por sus siglas en inglés) como un motor subyacente para comunicarse con los objetivos. XMD proporciona la interfaz de protocolo remoto GDB y sockets basados en TCP para las herramientas de depuración. También proporciona una interfaz basada en TCL para control y depuración del objetivo y para ejecutar instrucciones de prueba con verificación compleja para comprobar el sistema completo.

Capítulo IV

IV. DISCUSIÓN DE RESULTADOS

IV.1 INTRODUCCIÓN

La implementación del trabajo de tesis se realizó en la plataforma de evaluación ML403 la cual incluye un FPGA Virtex-4 FX y se implementaron los algoritmos de procesamiento digital de imágenes discutidas en el capítulo II. Se realizó el diseño de un sistema con procesador embebido utilizando el PowerPC 405 para el aprovechamiento máximo de los recursos del FPGA.

El controlador de video utilizado en el sistema con procesador embebido implementado en el FPGA Virtex-4 FX, alcanza una resolución máxima de 640x480 píxeles, por lo cual solo se utilizan imágenes de máximo dicha resolución, aunque la memoria RAM con la que se cuenta es de 64 Mb solo se necesitan 2 Mb para la memoria de video y 2 Mb para almacenar la imagen a procesar. Es importante tomar en cuenta que la frecuencia de reloj del procesador embebido PowerPC 405 utilizado en la implementación del sistema es de 300 MHz, lo cual es útil para realizar las comparaciones en el tiempo de procesamiento de los distintos algoritmos respuesta.

IV.2 COMPARACIÓN DE RESULTADOS

Se realizaron distintas pruebas del sistema con procesador embebido implementado en el FPGA Virtex-4 FX con imágenes de 640x480 píxeles en donde se analiza el tiempo de respuesta de los distintos algoritmos de procesamiento digital de imágenes.

Las mismas pruebas fueron realizadas a tres computadoras personales convencionales, la implementación de los algoritmos de procesamiento digital de imágenes en dichas computadoras se realizó para el sistema operativo Windows® y se utilizó el lenguaje de desarrollo Java 1.6.0 de Sun Microsystems®.

Las características de las computadoras personales son las siguientes, se incluye también las características del sistema con procesador embebido implementado en el FPGA Virtex-4 FX para propósitos de comparación:

	Sistema operativo	Procesador	Memoria
Implementación en el FPGA Vitex-4	Standalone	PowerPC®405, a 300MHz	64 Mb a 266 MHz
Computadora personal A	Windows Vista SP1	Intel Pentium Dual Core a 1.73 GHz	1 GB DDR2 a 667 MHz
Computadora Personal B	Windows XP SP2	Intel Pentium 4 a 2 GHz	2 GB DDR2
Computadora Personal C	Windows XP SP2	Intel Celeron a 634 MHz	512 Mb PC100

Tabla 1. Características de las distintas plataformas de desarrollo

Los resultados obtenidos al procesar una imagen de 640x480 pixeles en las distintas plataformas de desarrollo con los distintos algoritmos de procesamiento digital de imágenes fueron los siguientes:

	Negativo	Lupa de intensidades	Media Aritmética	Extracción de bordes	Suma de dos imágenes
Implementación en el FPGA Virtex-4	0.6 Segundos	1.1 segundos	0.9 Segundos	0.9 Segundos	0.9 Segundos
Computadora personal A	0.2 Segundos	0.4 segundos	0.3 Segundos	0.3 Segundos	0.3 Segundos
Computadora personal B	0.15 Segundos	0.4 segundos	0.25 Segundos	0.3 Segundos	0.25 Segundos
Computadora personal C	0.8 Segundos	1.2 segundos	1.1 Segundos	1 Segundo	1 Segundo

Tabla 2. Tiempos de procesamiento obtenidos al procesar una imagen de 640x480 pixeles en las distintas plataformas de desarrollo con los distintos algoritmos de procesamiento digital de imágenes.

Los resultados mostrados en la **Tabla 2** son resultados aproximados, se tomo el tiempo promedio de procesamiento en distintas pruebas realizadas con el mismo algoritmo.

IV.3 CONCLUSIONES

Tomando en cuenta la velocidad de procesamiento de los distintos procesadores de las computadoras personales las cuales son claramente superiores a la velocidad de procesamiento

del procesador embebido PowerPC 405 implementado en el FPGA Virtex-4 FX, el tiempo de respuesta obtenido en el sistema con procesador embebido implementado en el FPGA Virtex-4 FX muestra un rendimiento mayor al sistema implementado en una computadora personal.

En el peor de los casos el tiempo de respuesta del sistema con procesador embebido PowerPC 405 es solo 4 veces el tiempo de respuesta del procesador de la computadora personal B (Pentium 4 a 2 GHz), la velocidad del procesador de la computadora personal B es 6 mayor que la velocidad de procesamiento del procesador embebido PowerPC 405 a 300 MHz.

Analizando el caso de la computadora personal C mostrada en la **Tabla 1**, podemos observar que la velocidad del procesador Intel Celeron es de 634 MHz, lo cual es 2 veces la velocidad de procesamiento del procesador embebido PowerPC 405 a 300MHz, pero el tiempo de procesamiento de los distintos algoritmos de imágenes digitales es mayor en el procesador Intel Celeron comparado con el procesador embebido PowerPC 405.

Debido a que el sistema implementado en este trabajo de tesis es un sistema dedicado, muestra algunas ventajas en comparación con el uso de una computadora personal convencional, entre estas ventajas destacan las siguientes:

- Menor tiempo de procesamiento, como podemos ver en las tablas anteriores el tiempo de respuesta del sistema de procesador embebido implementado en el FPGA Virtex-4 FX es menor en comparación con el tiempo de respuesta del sistema implementado en una computadora personal tomando en cuenta que la velocidad de procesamiento del FPGA Virtex-4 es hasta 6 veces menor.
- Menor tamaño, el sistema final solo constara del FPGA Virtex-4 FX, la memoria Compact Flash, el cable de alimentación de energía eléctrica y la salida VGA. Esto se traduce en una tarjeta de tamaño no mayor a 10 centímetros de ancho por 10 centímetros de largo.
- Menor consumo de energía, debido a su menor tamaño y a su reducido número de periféricos, el consumo de energía es mucho menor que el de una computadora personal.
- Facilidad de uso, cuenta con una interfaz de usuario simple e intuitiva la cual permite una interacción con el usuario amigable y de fácil acceso.

La **Figura 40**, muestra la plataforma de evaluación ML403 ejecutando el sistema con procesador embebido PowerPC 405 implementado en el FPGA Virtex-4 FX, mostrando una imagen de 640x480 pixeles a través de la salida VGA.



Figura 40. Ejecución del sistema con procesador embebido PowerPC 405, mostrando una imagen a través de la salida VGA.

Capitulo V

V. CONCLUSIÓN

V.1 INTRODUCCIÓN

Para el desarrollo de este trabajo de tesis nos planteamos como objetivo principal el aumentar la velocidad de procesamiento de una imagen de mastografía utilizando hardware reconfigurable como plataforma de desarrollo.

Para alcanzar este objetivo, se realizo la implementación de distintas técnicas de procesamiento digital de imágenes en la plataforma de evaluación ML403, la cual incluye un FPGA Virtex-4 FX y nos permitió realizar el diseño de un sistema con procesador embebido, aprovechando al máximo los recursos del FPGA.

En distintas pruebas realizadas la implementación del sistema con procesador embebido PowerPC en un FPGA Virtex-4 FX ha alcanzado el objetivo planteado pudiendo acelerar el procesamiento de una imagen de mastografía y en general de cualquier imagen como radiografías, tomografías, entre otras.

V.2 TRABAJO A FUTURO

En este trabajo de tesis, existen grandes posibilidades de realizar contribuciones importantes en el campo de la detección temprana del cáncer de mama, así como de otras enfermedades o padecimientos utilizando plataformas de desarrollo de hardware reconfigurable. Este trabajo cuenta con muchas limitaciones, las cuales pueden ser eliminadas en futuros trabajos, a continuación se hace una breve descripción de estas posibles mejoras:

- Modificar la implementación de algunos algoritmos de imágenes digitales de forma tal que el procesamiento se realice en paralelo, esto reduciría en forma significativa el tiempo de respuesta del FPGA.
- Aprovechar al máximo los recursos del FPGA utilizando el enfoque Hardware-Software con lo cual los algoritmos complejos pueden implementarse en Hardware y los algoritmos simples o tareas simples en Software.

- Aumentar la resolución de salida del controlador de video hasta 1280x960, para aprovechar al máximo las características del FPGA Virtex-4 FX, actualmente la resolución máxima es de 640x480 pixeles.
- Implementar el uso de distintos formatos de imágenes como JPEG, GIF, entre otros, actualmente el sistema solo soporta el formato BMP.
- Implementar el uso de dispositivos de almacenamiento USB para la obtención de las imágenes, actualmente la obtención de las imágenes se realiza a través de una memoria Compact Flash.
- Acoplar el sistema actual para que realice procesamiento de video, lo cual ampliaría en gran medida utilidad del mismo.

BIBLIOGRAFÍA

Achronix Semiconductor Corporation. 2008. Our Company. [En línea] Abril de 2008. [Citado el: 1 de Abril de 2008.] <http://www.achronix.com/company.html>.

Altera. 2008. Technology Milestones. [En línea] Abril de 2008. [Citado el: 1 de Abril de 2008.] http://www.altera.com/corporate/about_us/history/abt-history.html.

Atmel. 2008. Corporate. [En línea] Abril de 2008. [Citado el: 1 de Abril de 2008.] <http://www.atmel.com/corporate/>.

Gonzalez, Rafael C. y Woods, Richard E. 2002. *Digital Image Processing*. Segunda edicion. s.l. : Prentice Hall, 2002.

Lattice Semiconductor Corporation. 2008. About Us. [En línea] Abril de 2008. [Citado el: 1 de Abril de 2008.] <http://www.latticesemi.com/corporate/>.

Martin, Manuel. 2008a. Notas de Imagenes Digitales, Capitulo 3. [En línea] Abril de 2008a. [Citado el: 1 de Abril de 2008.] <http://www.cs.buap.mx/~mmartin/pdi/PDI-Cap3.pdf>.

—. **2008b.** Notas de Imagenes Digitales, Capitulo 4. [En línea] Abril de 2008b. [Citado el: 1 de Abril de 2008.] <http://www.cs.buap.mx/~mmartin/pdi/PDI-Cap4.pdf>.

MathStar. 2008. Corporate Overview. [En línea] Abril de 2008. [Citado el: 1 de Abril de 2008.] <http://www.mathstar.com/Corporate.php>.

Microsoft Student 2008 [DVD]. 2007a. *Cáncer (medicina)*. s.l. : Microsoft Corporation, 2007a.

—. **2007b.** *Cáncer de mama*. s.l. : Microsoft Corporation, 2007b.

—. **2007c.** *Mamografía*. s.l. : Microsoft Corporation, 2007c.

Nájera, Manuel. 2007. *Tesis de Licenciatura, Software y Hardware para el análisis de mastografía.* Puebla : Benemerita Universidad Autonoma de Puebla, Facultad de Ciencias de la Computacion, 2007.

Pajares, Gonzalo y M. de la Cruz, Jesús. 2002. *Vision por Computador: imágenes digitales y aplicaciones.* Mexico : Alfaomega, 2002.

Pajares, Gonzalo, y otros. 2004. *Imagenes Digitales. Procesamiento practico con Java.* s.l. : Alfaomega, 2004.

QuickLogic. 2008. Corporate Overview. [En línea] Abril de 2008. [Citado el: 1 de Abril de 2008.] <http://www.quicklogic.com/home.asp?PageID=191&sMenuID=212&p1=212&p2=400>.

Texas Instruments. 2008. Is DSP Right for You? [En línea] Abril de 2008. [Citado el: 1 de Abril de 2008.] <http://focus.ti.com/dsp/docs/dspsupporto.tsp?sectionId=4&tabId=1443>.

Xilinx, Inc. 2008a. Getting Started with FPGAs. [En línea] Abril de 2008a. [Citado el: 1 de Abril de 2008.] <http://www.xilinx.com/company/gettingstarted/index.htm>.

—. **2006.** *ML401/ML402/ML403 Evaluation Platform: User Guide.* s.l. : Xilinx, 2006.

—. **2008b.** Our History. [En línea] Abril de 2008b. [Citado el: 1 de Abril de 2008.]

<http://www.xilinx.com/company/history.htm>.

—. **2007.** *Platform Studio Help.* s.l. : Xilinx, 2007.

—. **2008d.** Spartan-3 Generation FPGAs. [En línea] Abril de 2008d. [Citado el: 1 de Abril de 2008.] http://www.xilinx.com/products/silicon_solutions/fpgas/spartan_series/index.htm.

—. **2008c.** Virtex Series. [En línea] Abril de 2008c. [Citado el: 1 de Abril de 2008.]

http://www.xilinx.com/products/silicon_solutions/fpgas/virtex/index.htm.