

Capítulo 6

Pruebas y Resultados

6.1 Pruebas con cámara CCD.

Se construyó una base provisional con el objetivo de mantener constante las condiciones del ambiente luminoso en el que se realizarían las pruebas.

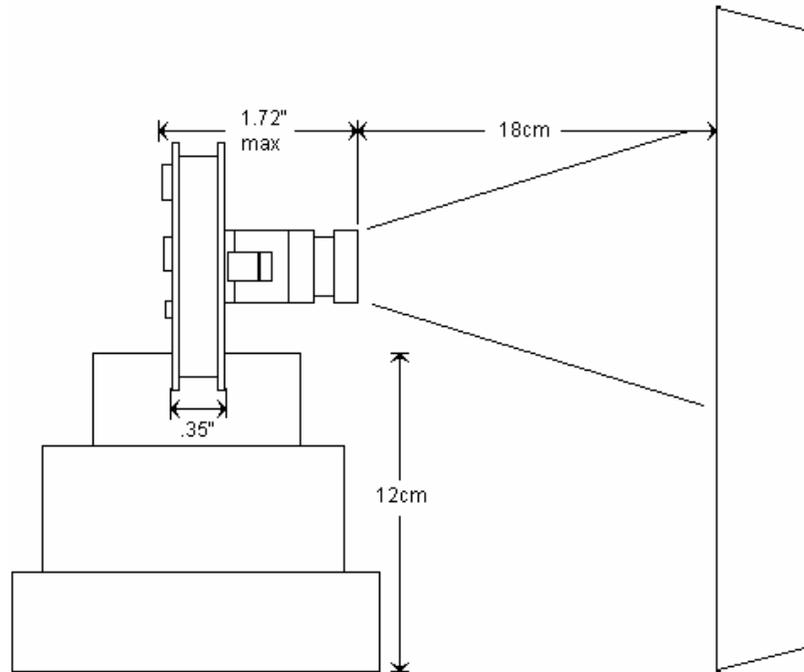
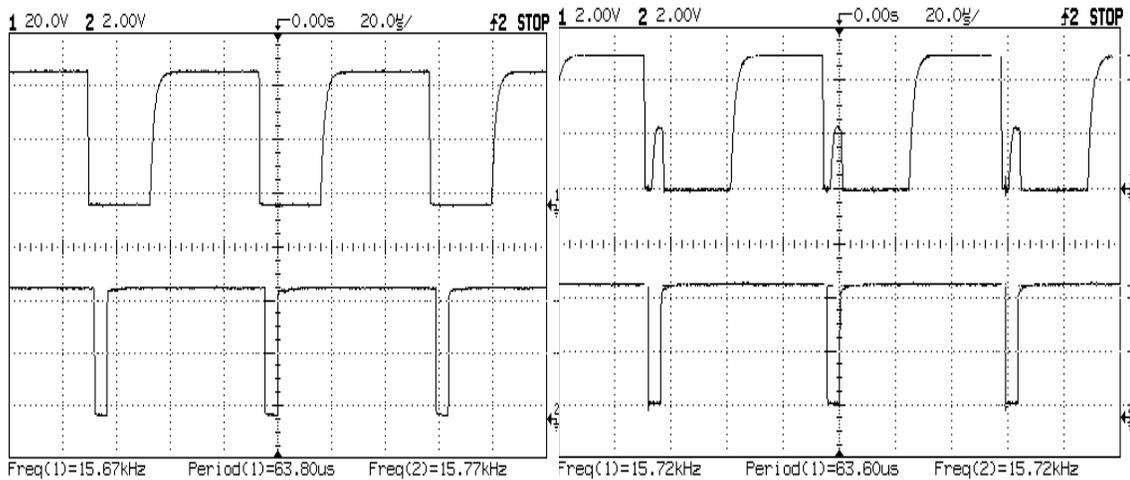


Figura 6.1 Diagrama de cámara CCD a distancia.

En la figura 6.1 se muestra un diseño ideal para el posicionamiento de la cámara CCD, después de que se hicieron una serie de pruebas se llegó a una distancia prudente en la que la lente de la cámara enfocaba las imágenes sin ningún problema, 18 cm. aproximadamente fue lo que se calculó.

6.2 Interfaz.

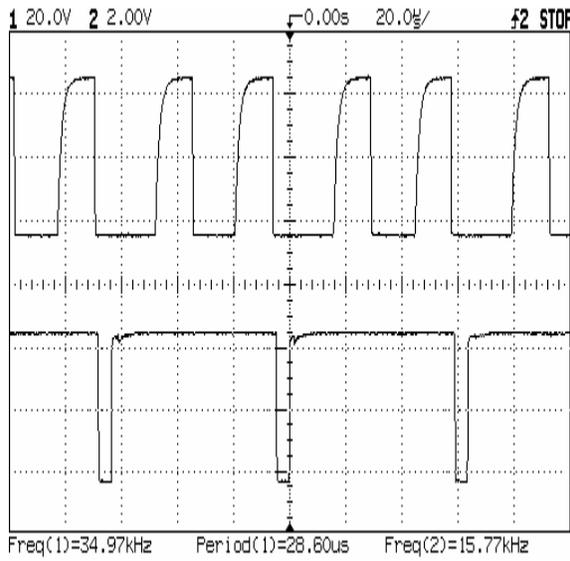
Durante las pruebas realizadas a los componentes que conforman la interfaz se fue descubriendo que los dispositivos operacionales necesitan un ancho de banda más grande de lo normal, esto debido a que el ancho de banda de la señal de video es de 4 MHz aproximadamente (sin sonido ni datos). Este proyecto se basó en el uso del sistema americano y se encontraron distintas anomalías con los amplificadores operacionales LM741; a continuación se presentan una serie de gráficas que muestran las diferencias encontradas entre los dos tipos de operacionales.



Figuras 6.2 (LF347) Detecta Blanco.

Figura 6.3(LM741) Detecta Blanco.

Se puede notar que la señal #1 es la salida del comparador mandando solo 1 lógico al FPGA y la señal #2 indica cual es la salida de sincronía compuesta, la figura 6.2 contó con el amplificador LF347 que tiene un ancho de banda de 4 MHz, es de apreciar que su salida de comparador es muy “limpia”, mientras que en la figura 6.3 contó con el amplificador operacional LM741 (popular), pero su ancho de banda es de tan solo 1 MHz lo cual explica el por que de ese pequeño pico a la salida de su comparador.



Figuras 6.4 (LF347) 1/2blanco y 1/2negro.

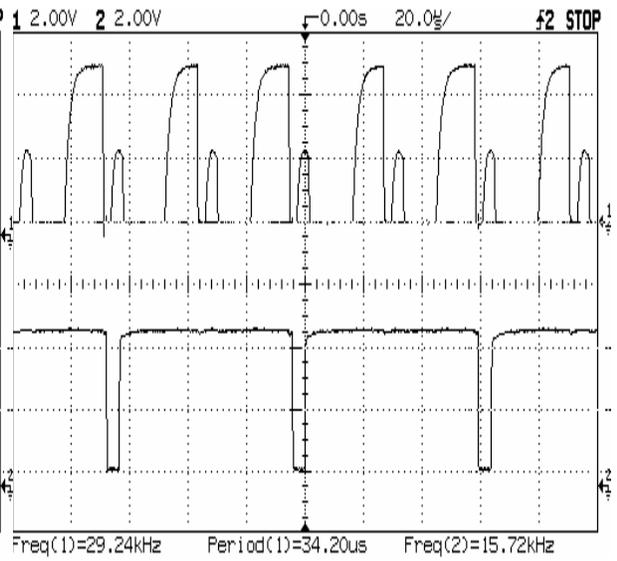


Figura 6.5 (LM741) 1/2blanco y 1/2negro.

De igual forma que en las señales anteriores se puede observar la diferencia entre el amplificador operacional LF347 y el LM741, debido a ese pequeño pico en la entrada del FPGA se creaba interferencia que hacía dudar de la precisión de los módulos dentro de la implementación del diseño. Las figuras 6.6 y 6.7 se muestran el color negro y su diferencia.

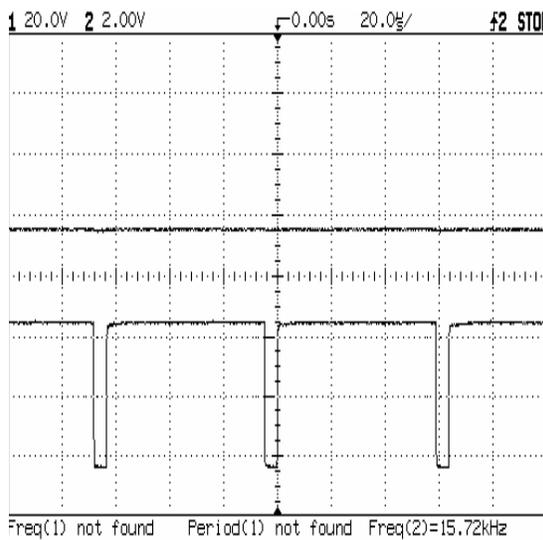


Figura 6.6 (LF347) Color Negro.

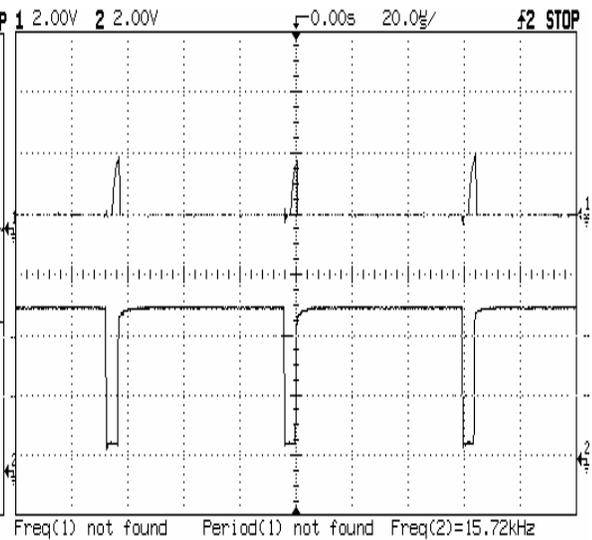


Figura 6.7 (LM741) Color Negro.

6.3 Identificación del camino trazado.

En resumen el programa básicamente hace el conteo de filas y columnas de acuerdo a la combinación de señales que controlan cada uno de los módulos, que en su mayoría están compuestos por registros que atrapan los datos de acuerdo a la configuración de condiciones dadas por las señales, en la figura 5.10 del capítulo 5 se encuentran bien detallados dichos módulos y procesos que realizan todo el análisis de píxeles blancos y negros.

El código fuente del programa en VHDL se encuentra en el apéndice A, y en este se encuentra la descripción paso a paso de cada uno de los módulos y procesos para atrapar la información requerida

MUESTRAS = 400

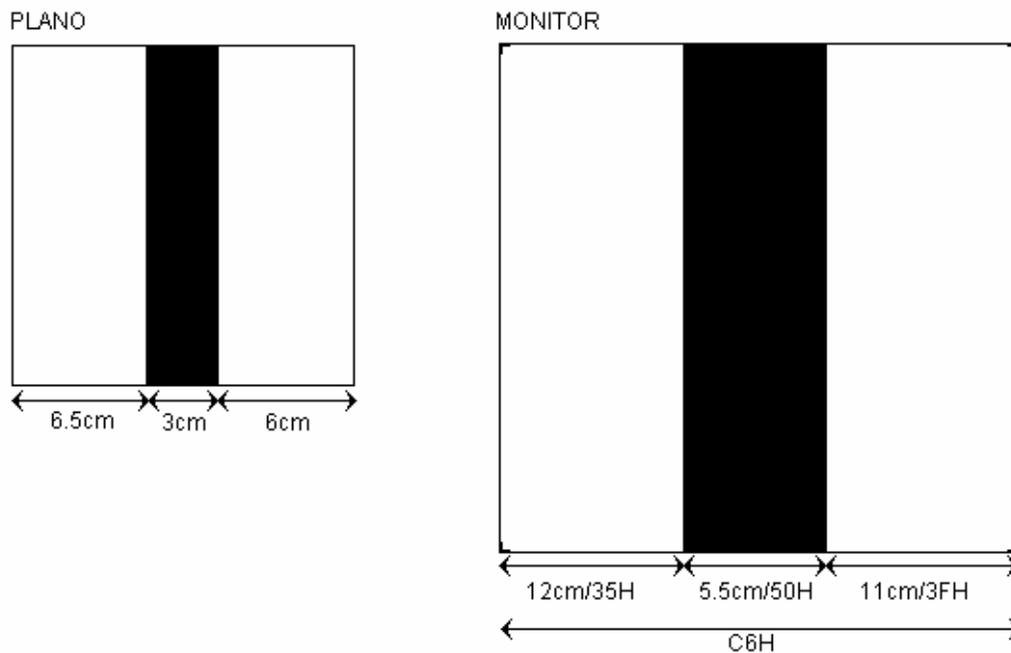


Figura 6.8 Ejemplo de prueba realizada de detección de blanco y negro.

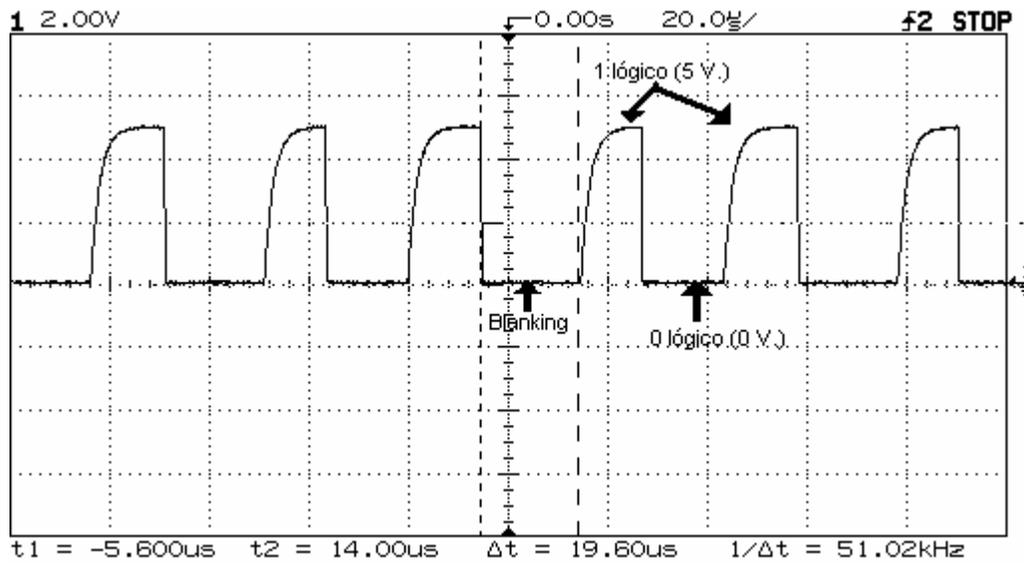


Figura 6.9 Gráfica de salida del comparador (LM339).

Como se puede apreciar en la figura 6.12 la salida del comparador coincide con la imagen tomada por la cámara CCD reflejada en el monitor de la figura 6.11, es una demostración de la precisión que se puede obtener durante las pruebas.

6.4 Programación y FPGA.

Debido a las constantes pruebas a las que se tenía que someter cada fase del proyecto, se crearon una gran cantidad de programas tanto en software como en hardware para filtrar cualquier problema que estuviera causando conflicto en algún dispositivo no considerado.

Un problema que se presentó fue que el circuito en general no quería hacer el conteo de Líneas debido a que la señal que venia de la interfaz al parecer no estaba funcionando, se creó una clase de simulador físico con otro FPGA para saber si:

- El componente (LM1881) de la interfaz no era compatible con el FPGA
- La señal no cumplía los requerimientos para inicializar el contador.
- O bien el programa implementado no estaba funcionando a pesar de que todo saliera bien en la simulación.

El FPGA utilizado para hacer esta tarea fue el DIGILAB2XL que es una versión más pequeña que el DIGILAB2

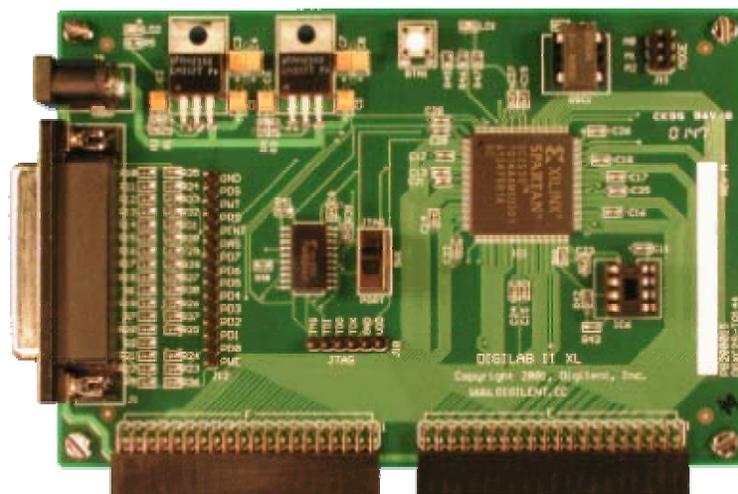


Figura 6.10 DIGILAB2XL.

Se generó un pulso que fuera lo mas cercano posible a la señal de Línea (generada por la interfaz) para probar si en realidad estaba funcionando correctamente.

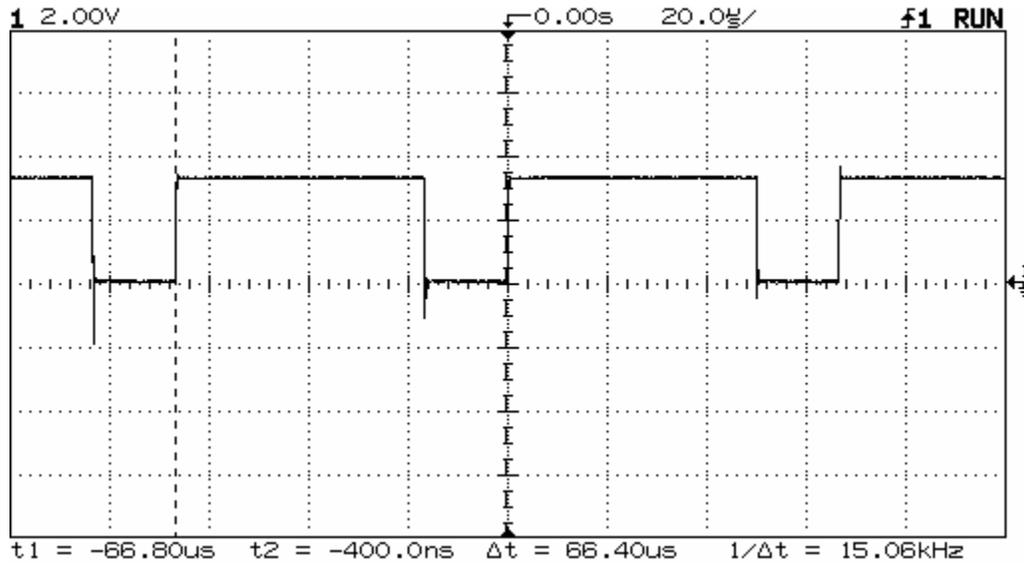


Figura 6.11 Señal de Línea simulada en otro FPGA.

Al término de las pruebas se pudo encontrar que uno de los pines disponibles para introducir una de las señales, en este caso la de Línea no funcionaba del todo bien ya que este se encuentra compartiendo la conexión con otro dispositivo, entonces se reubicó esta señal en otro pin disponible.

6.5 Programación.

La simulación de los dispositivos (módulos y procesos) programados en el software ACTIVE-HDL, son de gran ayuda debido a que el usuario puede dar tiempos predeterminados o virtuales para observar si hay un correcto funcionamiento en la lógica de programación. Como se puede apreciar en la figura 6.12 las salidas indicadas con una flecha muestran el cumplimiento de las condiciones que gobiernan a los registros que se dieron de alta en la programación del contador de píxeles blancos y negros los cuales son fácilmente identificables en la figura 6.12.

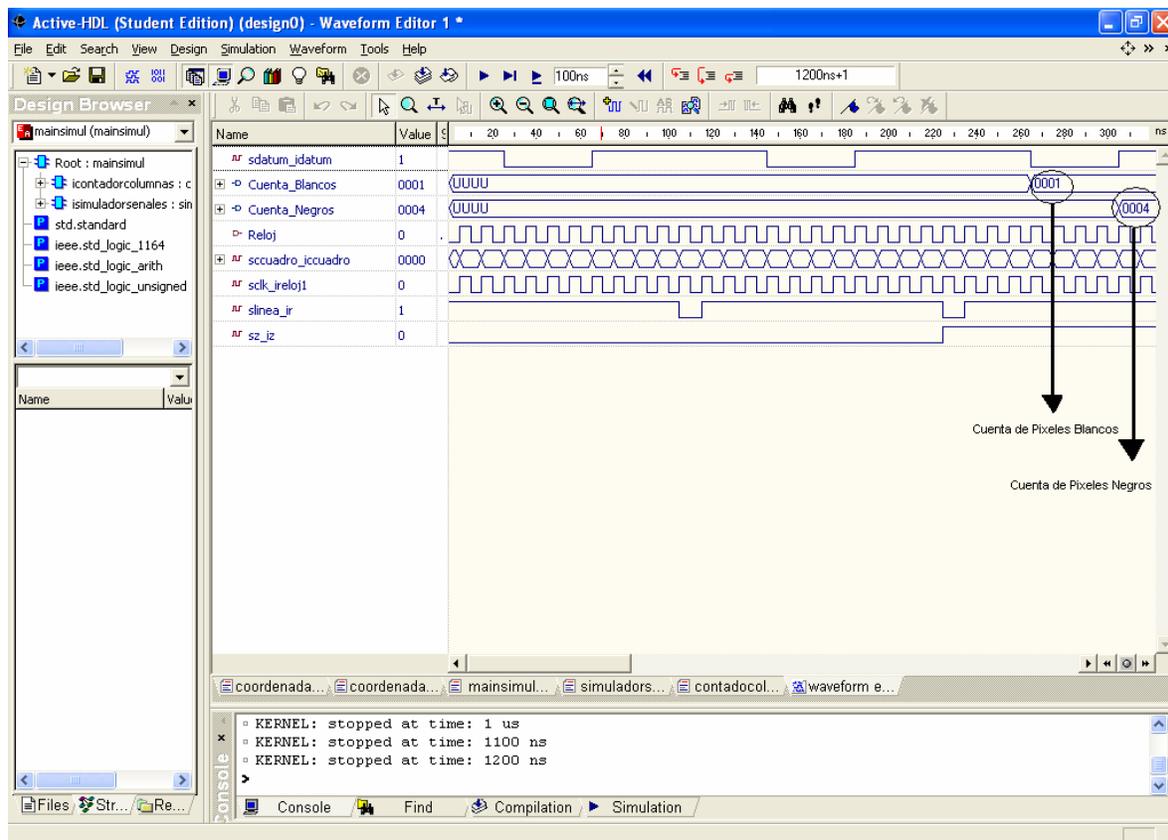


Figura 6.12 Simulación en ACTIVE-HDL del conteo de píxeles.

Obviamente al igual que en todo el proceso de pruebas y resultados, se tuvo que generar un diseño digital que fuera capaz de emular las señales generadas por el analizador de imágenes (adquisición de imagen).

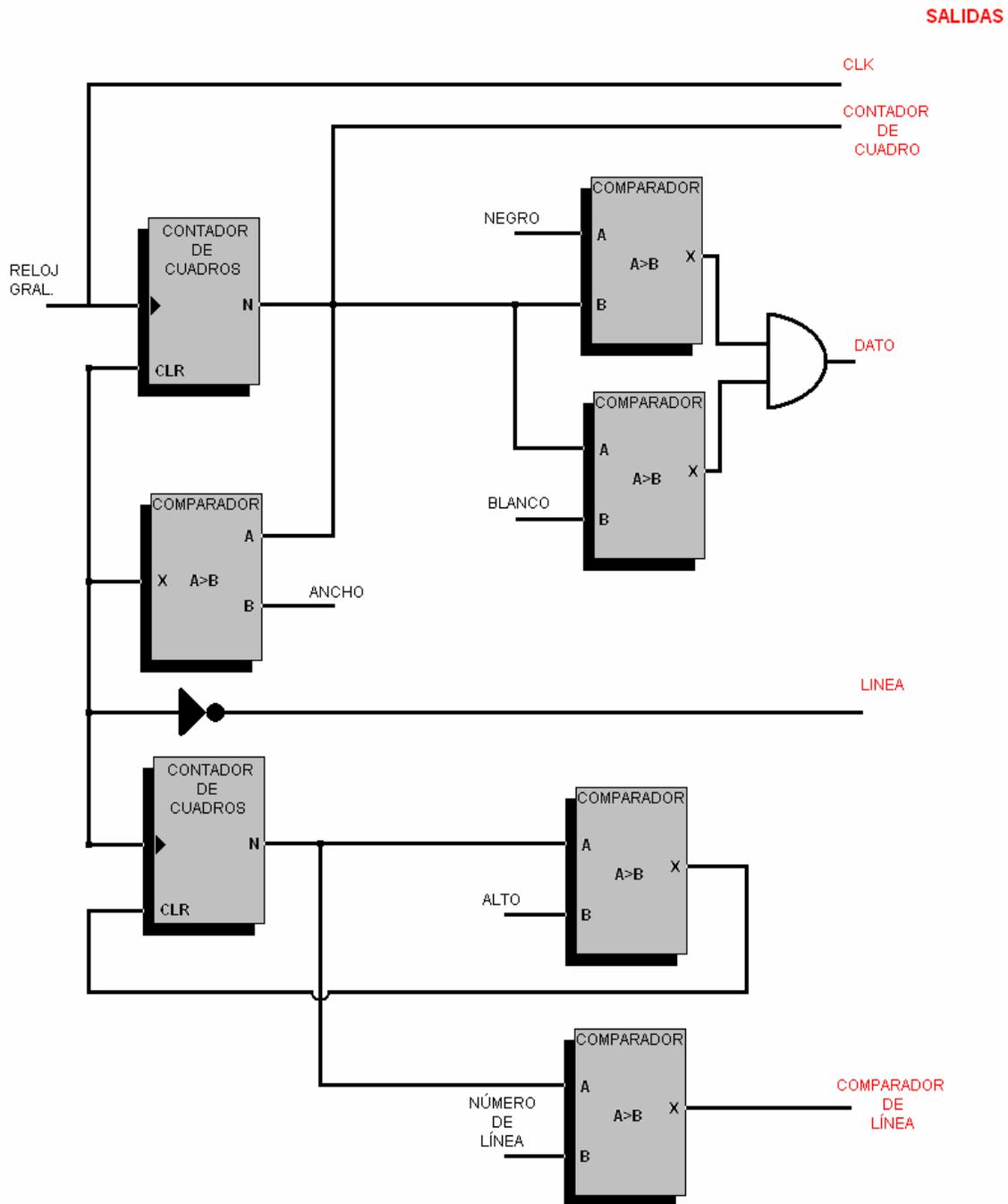
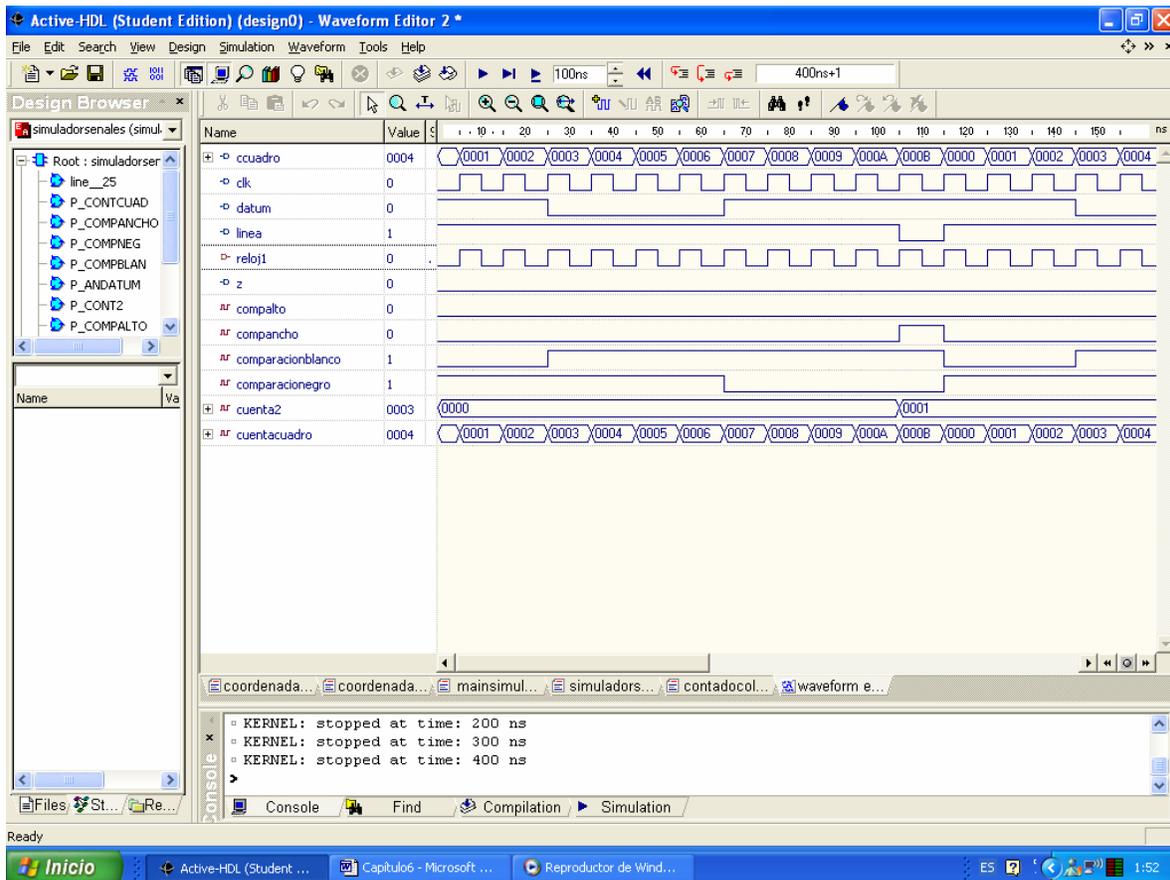


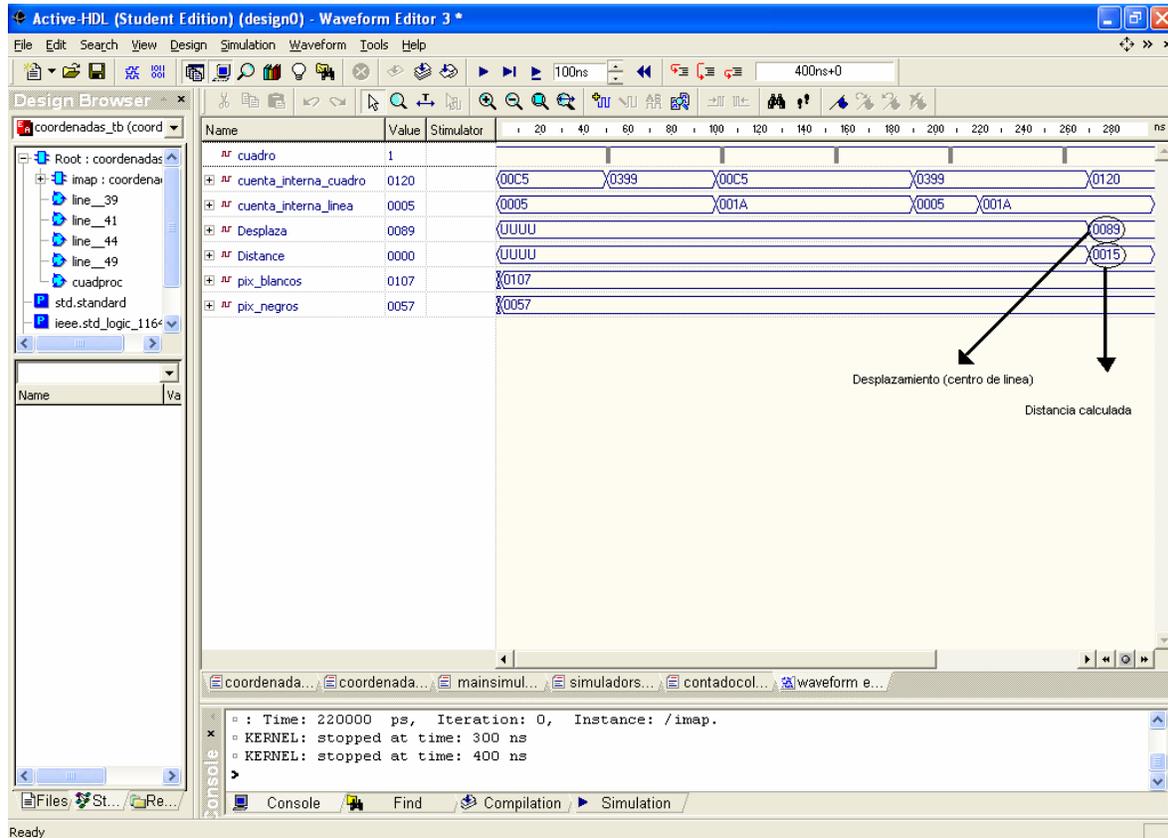
Figura 6.13 Simulador para respuesta de diseño de conteo de píxeles blancos y negros.



6.14 Simulación en ACTIVE-HDL del generador de señales para conteo de pixeles.

Para hacer pruebas más convincentes en el desarrollo de la programación se crearon algunos algoritmos, para tener la certeza de que están funcionando adecuadamente las señales que condicionan la acción de los registros y módulos que se presentan en la figura 6.13, de la misma forma se crearon mesas de trabajo o “Test bench” (Apéndice A) para probar cada uno de los dispositivos del diseño digital de análisis de imagen, para después verlos en una pantalla de graficación como se muestra en la figura 6.14.

6.6 Detección de Coordenadas.



6.15 Simulación en ACTIVE-HDL para detectar coordenadas.

La simulación mostrada en la figura 6.15 es una clara prueba de que los dispositivos tienen resultados positivos en tiempo virtual, lo cual indica que el uso de distintas herramientas (en este proyecto fue el ACTIVE-HDL) ayuda al entendimiento de la implementación de los diseños digitales en el FPGA.