

DISEÑO DE UN ANALIZADOR LÓGICO EN VHDL Y SU IMPLEMENTACIÓN EN UN FPGA SPARTAN 3

Michael Alejandro Diaz Illa^{1,2}, Alfredo Granados Ly¹

¹Universidad Nacional Mayor de San Marcos, Lima-Perú.

²Grupo de Microelectrónica PUCP, Lima-Perú.

mdiaz1@unmsm.edu.pe , agranadosl@unmsm.edu.pe

RESUMEN

Antiguamente el diseño e implementación de un sistema digital complejo era muy tedioso y podría llevar mucho tiempo desde su inicio hasta su culminación. Hoy en día con la aparición de las herramientas de síntesis y simulación los ingenieros especializados pueden terminar un diseño en corto tiempo disminuyendo costos y aumentando la productividad. En este trabajo se muestra el diseño de un analizador lógico implementado en un FPGA utilizando como entrada de diseño VHDL. La herramienta de síntesis y simulación que se utilizó fue el ISE 7.1 y pertenece a la empresa líder en lógica programable XILINX. Entre sus características podemos mencionar que este sistema puede tomar muestras de un máximo de 64 canales con una frecuencia de muestreo de hasta 50 MHz y se muestra en la figura 1. Como interfaz gráfica se ha usado un monitor VGA donde se visualiza las señales capturadas y para la configuración del analizador se ha utilizado un teclado.

1. INTRODUCCIÓN

Los analizadores lógicos hoy en día son instrumentos de gran necesidad en instituciones educativas, laboratorios de investigación avanzada y empresas privadas que realizan desarrollo electrónico, debido al gran desarrollo de los sistemas digitales. Por otro lado el alto costo de estos instrumentos nos ha llevado a pensar que un diseño e implementación en FPGA de uno de estos equipos causaría un ahorro en estas instituciones. A la hora de realizar el estudio de funcionamiento de sistemas digitales se pueden elegir como herramienta de análisis osciloscopios digitales DSO (digital storage oscilloscopes) o bien analizadores lógicos. El osciloscopio es un equipo muy familiar que suele ser muy útil en determinadas aplicaciones donde existen pocas señales de interés (2 ó 4 como máximo). Sin embargo, cuando se necesita realizar el análisis simultáneo de un gran número de señales digitales, o cuando se requiere de un sistema complejo de disparo ligado a un determinado patrón establecido a partir de múltiples señales digitales no siendo exigible una gran

exactitud en las medidas de amplitud y tiempo, es recomendable el uso de los analizadores lógicos.

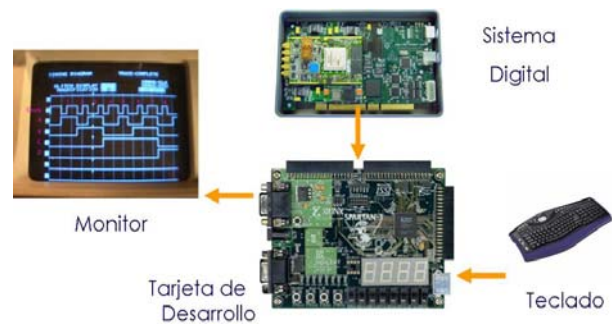


Figura 1: Analizador lógico.

2. DISEÑO DEL ANALIZADOR LÓGICO

El trabajo se ha dividido en el diseño de cuatro partes llamados: módulo de visualizar pantalla, módulo de pintar pantalla, módulo de captura y escritura de datos y un módulo de configuración. La implementación se ha realizado en la tarjeta de desarrollo SPARTAN 3 de la empresa XILINX de la cual se ha usado la salida VGA para el monitor, el dispositivo FPGA donde reside todo el diseño, el puerto PS/2 para la configuración del sistema a través de un teclado y la memoria externa SRAM para almacenar los datos que se imprimen en pantalla del monitor [1].

2.1. Módulo de visualizar pantalla y módulo de pintar pantalla

El módulo de visualizar pantalla consiste de un controlador de memoria que envía los datos de la SRAM del módulo de desarrollo SPARTAN3 a un generador de sincronismo con lo cual visualizamos los datos de la memoria en el monitor VGA. La figura 2 muestra el generador de sincronismo, básicamente consiste de un generador de sincronismo horizontal y otro vertical. Estos dos generadores deben generar señales de acuerdo a la frecuencia y resolución del monitor.

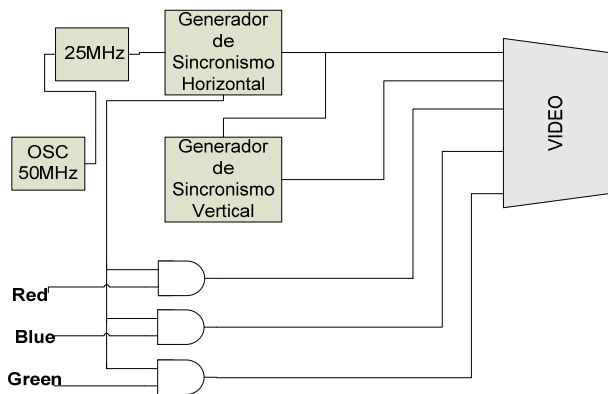


Figura 2: Generador de sincronismo.

La memoria SRAM que posee el módulo de desarrollo tiene 16 bits de datos de los cuales se han usado solamente los 12 bits menos significativos, por lo que en cada dirección se tiene cuatro píxeles en formato RGB. Los sincronismos vertical y horizontal que son contadores sirven para dar la dirección al controlador de memoria y luego este toma los datos de la memoria y los lleva a las entradas RGB del módulo de sincronismo. De esta manera el módulo lee las porciones de memoria y envía los datos (píxeles) al monitor. Se han usado 76800 direcciones de la memoria externa para cubrir los 640X480 píxeles del monitor. El módulo de pintar pantalla tiene como función inicializar la pantalla con un solo color para lo cual se escribe en las posiciones de la memoria externa cualquiera de las ocho combinaciones de colores RGB (negro, azul, verde, cyan, rojo, violeta, amarillo y blanco). Estos colores se seleccionan con tres switches de la tarjeta de desarrollo.

2.2. Módulo de captura y escritura de datos

Consiste de un circuito que captura las muestras (señales digitales de un sistema digital) con cierta frecuencia de muestreo durante un tiempo y se muestra en la figura 3. Las muestras son almacenadas en las memorias embebidas del FPGA.

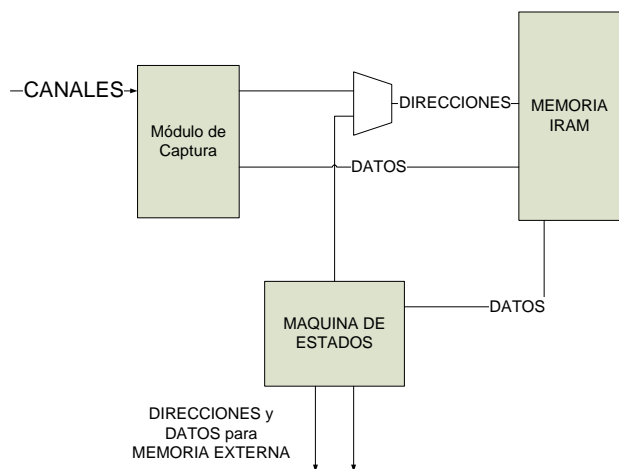


Figura 3: Módulo de captura y escritura de datos.

Cuando mayor sean los canales de muestreo se tendrán menos muestras, tales el caso que para 8 canales se pueden tener 24,564 muestras en un tiempo de 491,28 useg, para 64 canales se pueden tener 3070 muestras en un tiempo de 61,43 useg con una frecuencia de muestreo de 50MHz. Este módulo fue diseñado con dos maquinas de estado, una de ellas se encarga de tomar las muestras y almacenarlas en las memorias embebidas del FPGA y la otra maquina de extraer estos datos y llevarlos a la memoria SRAM que posee el módulo de desarrollo.

2.3. Módulo de configuración

Este módulo consiste de un bloque que recibe códigos del teclado y los codifica para llevarlos a una maquina de estados y se muestra en la figura 4. Este módulo sirve para configurar la frecuencia de muestreo, tiempo total de muestreo, el número de canales y también nos permite escribir en pantalla con un teclado.

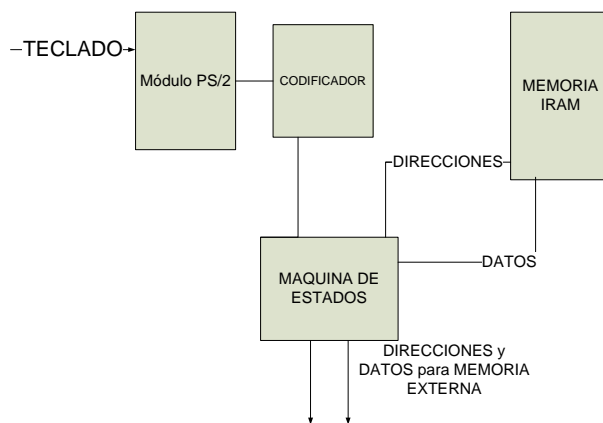


Figura 4: Módulo de configuración.

3. CONCLUSIONES

Se han usado un total de 120 CLB's en todo el diseño, vistos en el programa ISE 7.1 lo que equivale a la cuarta parte del FPGA XC3S200 [2]. Se realizaron pruebas con un circuito digital de prueba de ocho salidas. Este circuito se unió con el analizador lógico diseñado lográndose buenos resultados.

4. REFERENCES

- [1] "Spartan-3 Starter Kit Board User Guide." <http://www.xilinx.com/bvdocs/userguides/ug130.pdf> May 2005.
- [2] "Using the ISE Design Tools for Spartan 3." <http://www.xilinx.com/bvdocs/appnotes/xapp473.pdf> , May 2005.