



Trabajo práctico final

Sistemas digitales 66.17

Cátedra Alvarez

Segundo cuatrimestre de 2024

Alumno:	Henseler, Santiago
Padrón:	110732
Email:	shenseler@fi.uba.ar



Índice

Objetivos.....	3
Desarrollo.....	3
RAM.....	3
UART.....	4
Rotación 3D.....	4
Driver.....	5
Utilización de recursos.....	6

1. Objetivo

El objetivo del presente trabajo práctico es desarrollar una arquitectura de rotación de objetos 3D basada en el algoritmo CORDIC, implementando tanto la unidad aritmética de cálculo como así también un controlador de vídeo asociado VGA. Para realizar el trabajo práctico se utilizó el siguiente diseño propuesto por la cátedra:

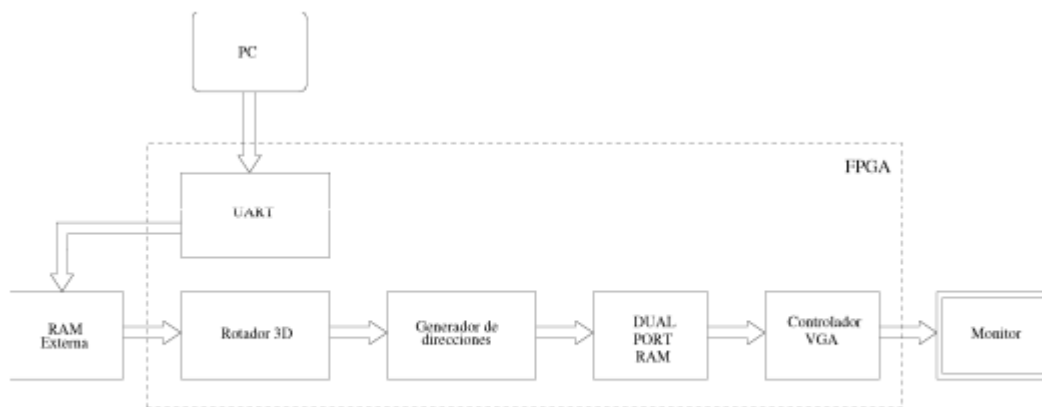


Figura 1. Esquema completo del sistema

Con la excepción de que al no contar con la FPGA de forma física la RAM externa fue reemplazada por las block-RAM del mismo.

2. Desarrollo

2.1 Memoria RAM

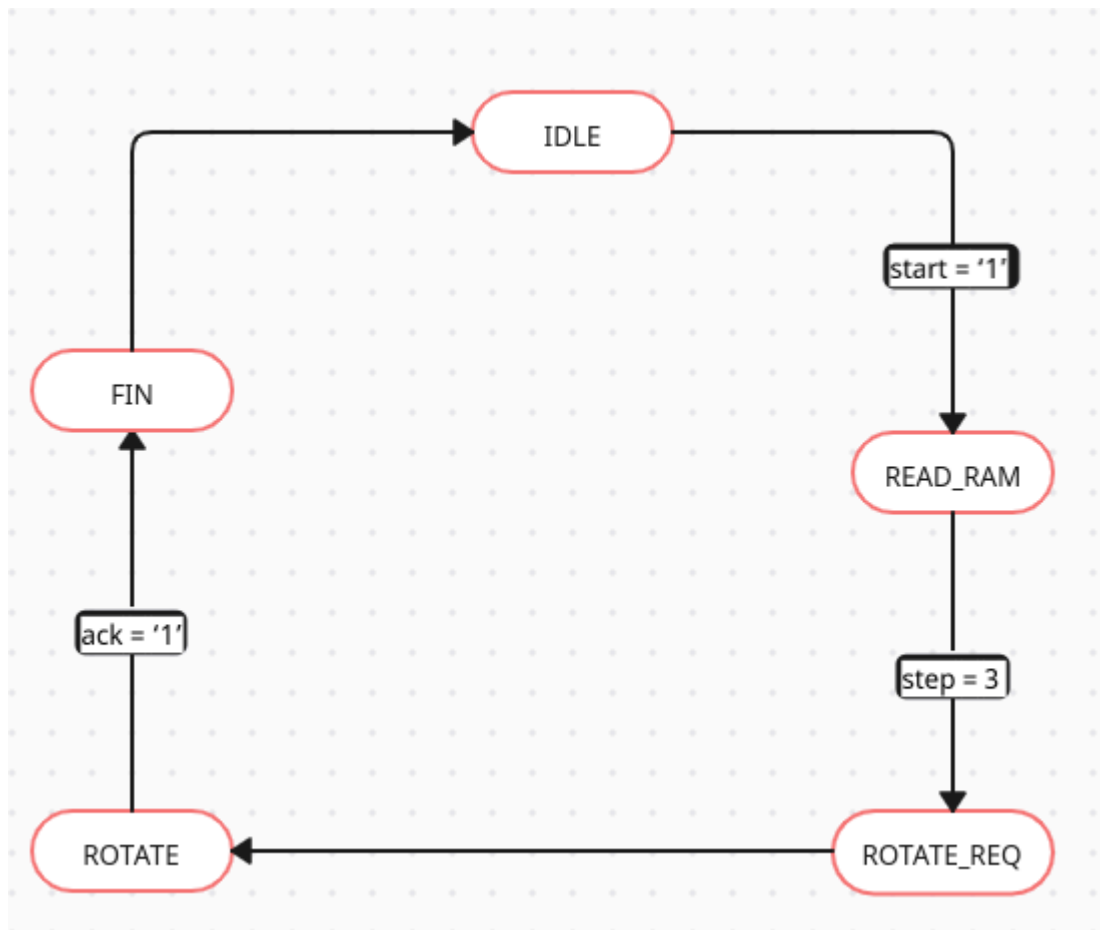
Al momento de implementar las memorias RAM se utilizó la descripción hecha por xiling en su manual para facilitar al sintetizador interpretar de la utilización de las block-RAM de la FPGA. Para el diseño propuesto se utilizaron 2 memorias rams. La primera almacena las coordenadas recibidas por el UART y tiene una capacidad de redireccionamiento de 65.536 posiciones con 8 bits por dirección. Dado que el archivo con las coordenadas del mundo tiene 11.946 vectores y por cada vector hay 3 enteros de 8 bits se termina utilizando 35.841 posiciones de la RAM. La segunda memoria RAM va a ser la encargada de almacenar los bits a ser mostrados por pantalla a través del VGA. Esta va a tener una capacidad de redireccionamiento de 524.288 posiciones de un bit cada una, porque la imagen a ser mostrada es en blanco y negro, de las cuales se utilizaran 307.200 (640X480) posiciones. En la memoria RAM de video se podría haber aplicado una optimización para reducir el tamaño de la misma ya que el mundito no ocupa todo el ancho de la pantalla.

2.2 UART

Al momento de recibir las coordenadas del mundo se utilizó un UART solo de recepción configurado con un baud rate de 115200 bps, 8 bits por dato y sin bit de paridad. Una vez confirmada la recepción del dato por parte del UART (`rx_data_rdy = '1'`) el controlador se encarga de almacenarlo en la primera memoria RAM de forma secuencial.

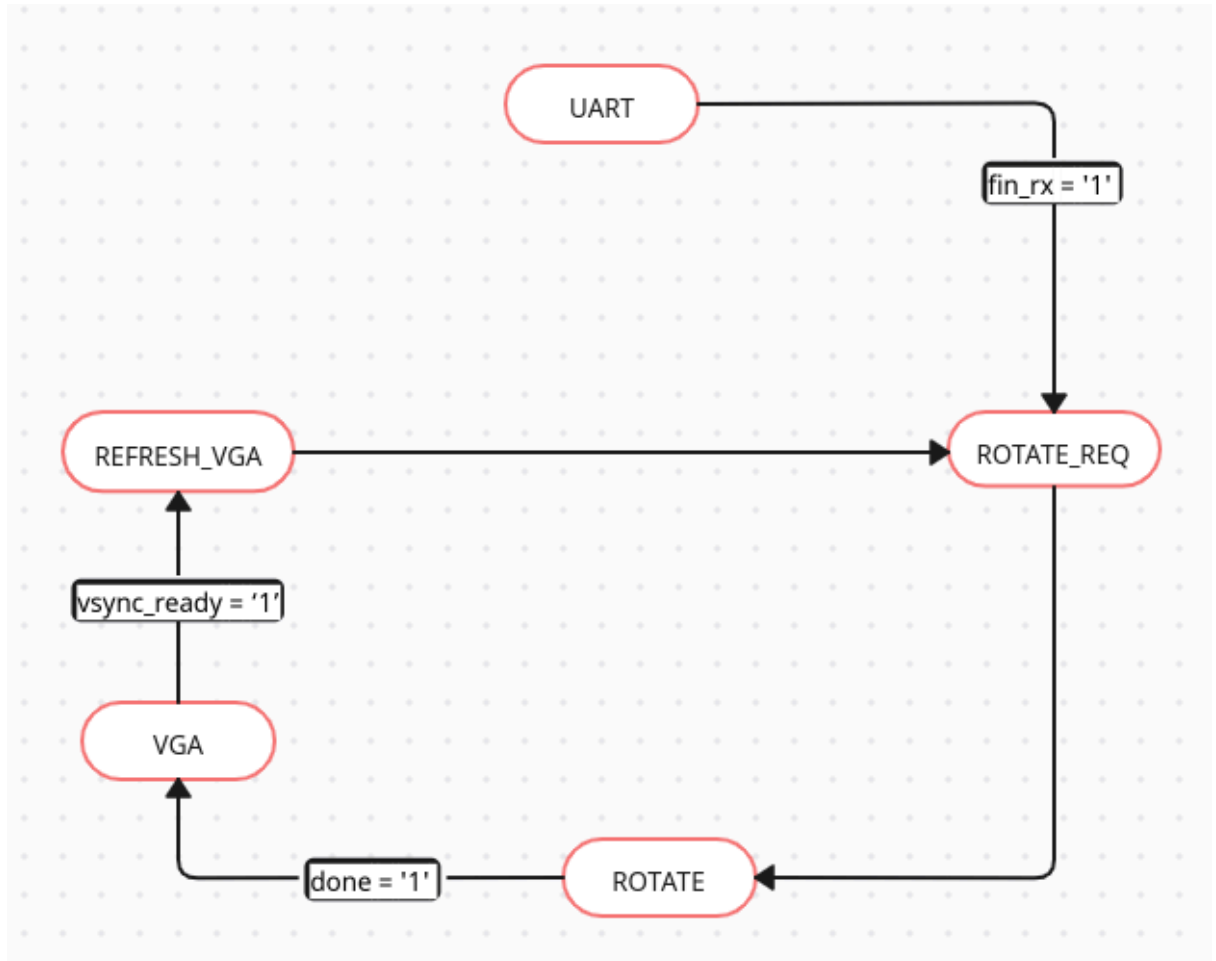
2.3 Rotación 3D

Una vez confirmada la recepción de todas las coordenadas por parte del controlador del UART (`fin_rx = '1'`) se activa la máquina de estados que se encarga de la rotación 3D. La máquina de estados es la siguiente:



2.4 Driver

Esta entidad es la encargada de controlar el funcionamiento de todo el dispositivo y hacer las interconexiones entre módulos. Para implementarla se utilizó la siguiente máquina de estados:



3. Utilización de recursos

