

Diseño, verificación y validación de sistemas digitales.

Practica 3: Acelerador de hardware para la operación de matriz por vector.

Abisai Ramírez

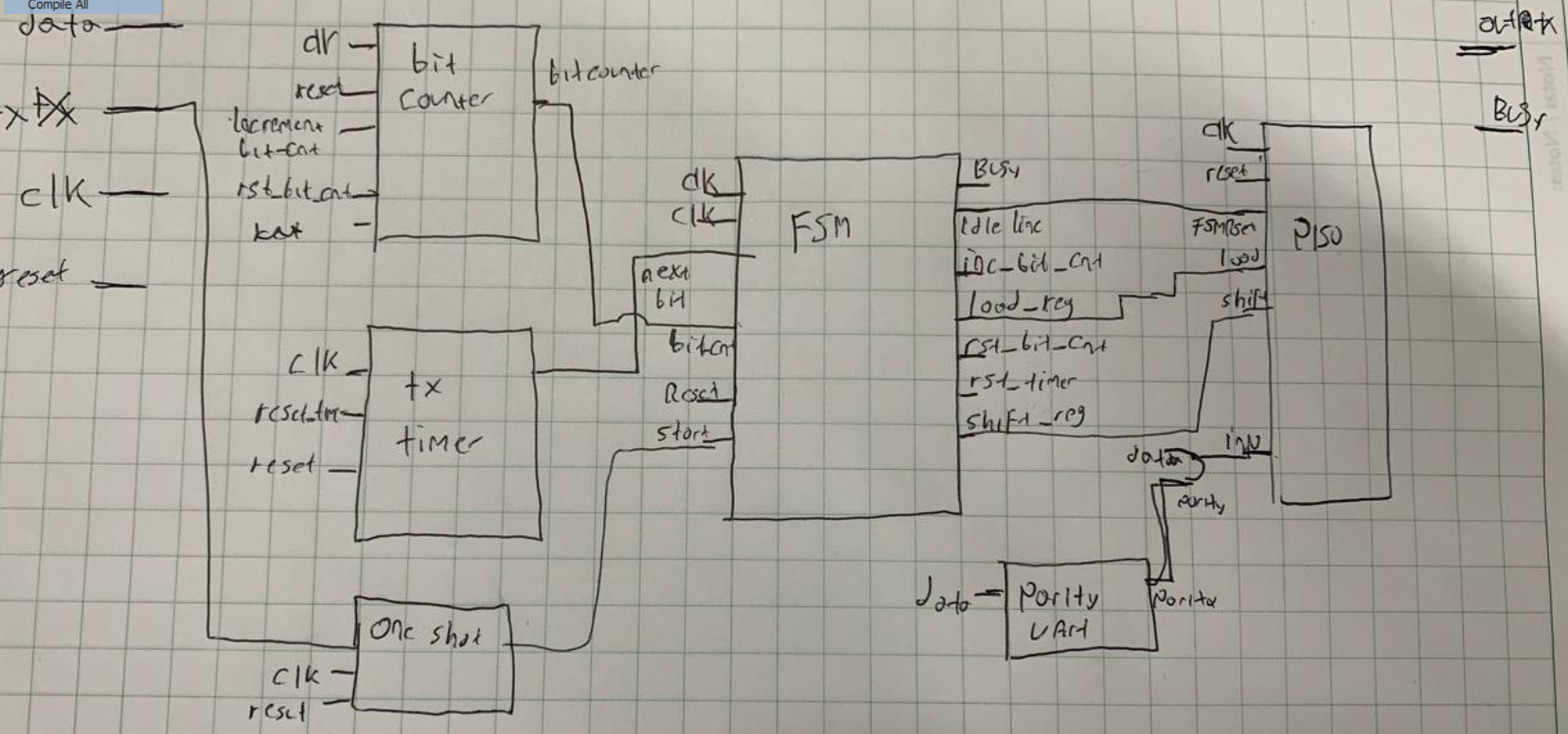
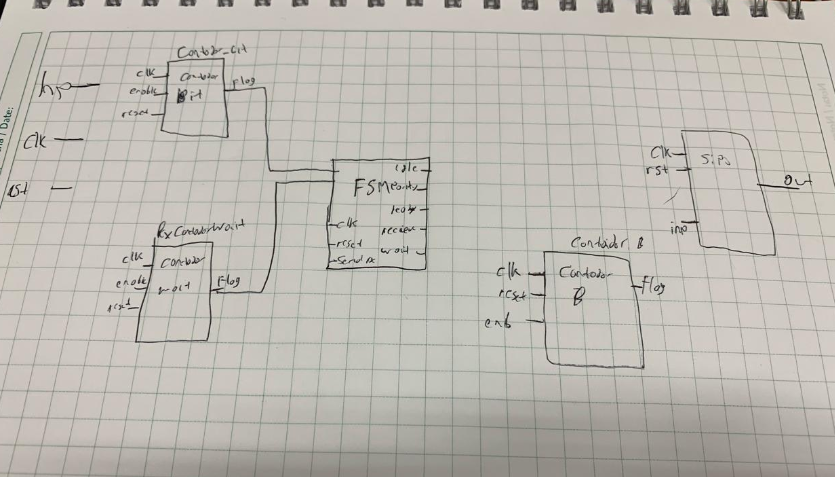
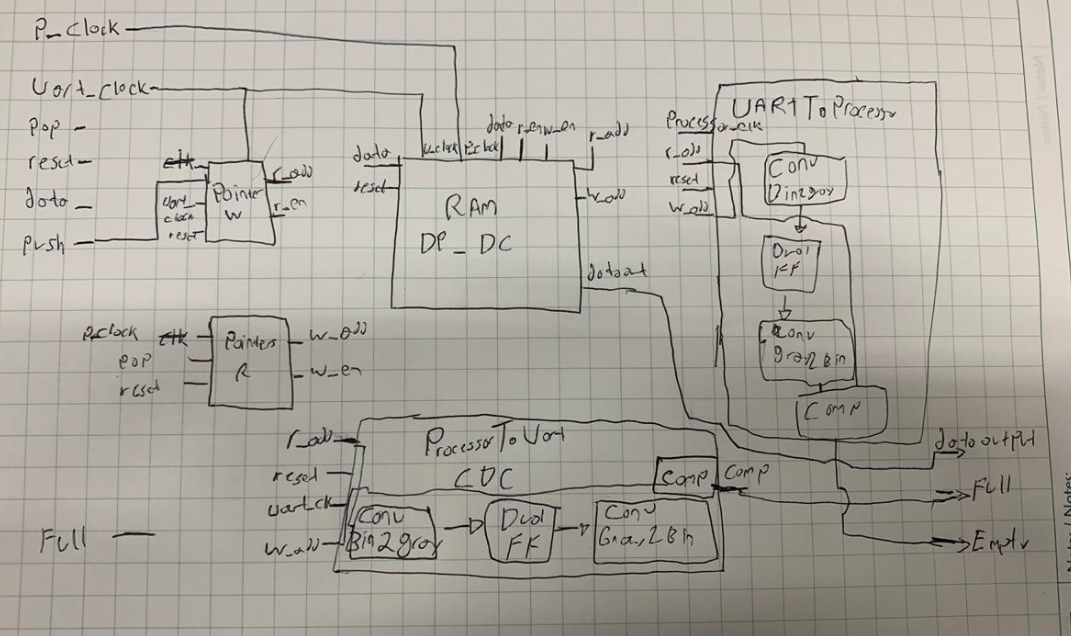
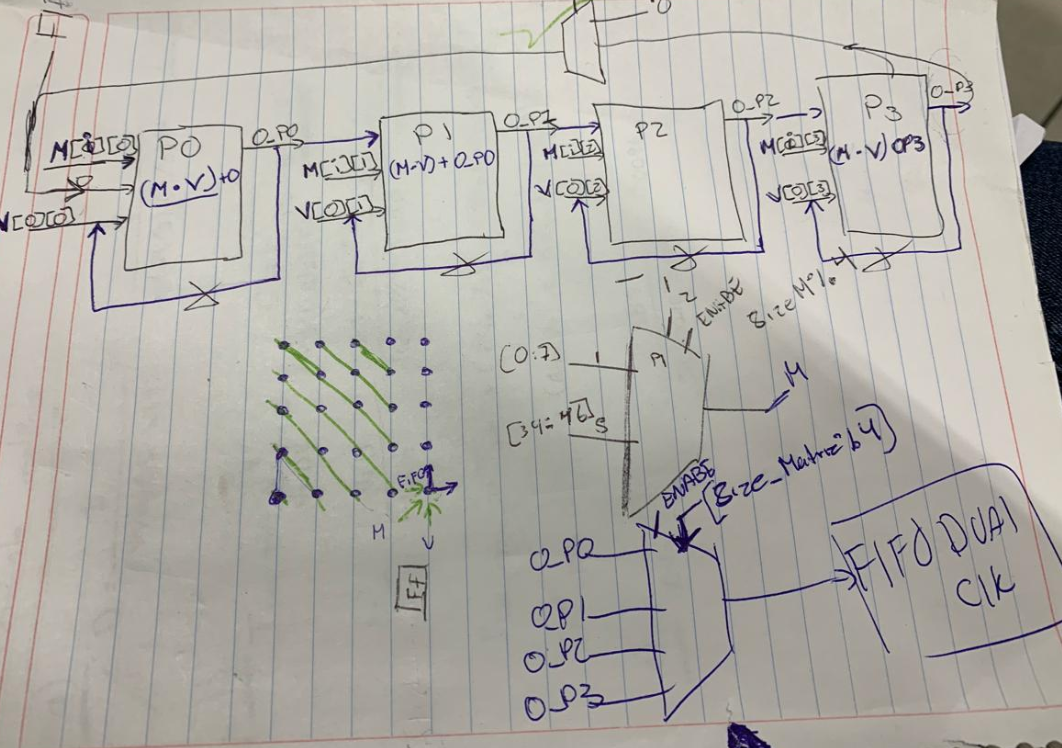
Integrantes:

Alberto Contreras.

Eric Guedea.

Introducción: Para la realización de esta practica se necesita crear un acelerador de hardware capaz de realizar operaciones de hasta 8x8 adaptando el ciclo de reloj a la longitud de la operación, la entrada de datos para el acelerador se realiza por medio de una UART, la cual al estar en dominios de reloj diferentes que el acelerador pasa los datos por medio de una fifo circular y un proceso de cruce de dominio de reloj tanto de entrada como de salida.

Modulos utilizados y diseño aproximado:

* UART TX y RX
  + 
  + 
* CDC Circular Fifo
  + 
* Procesador
  + 

Desarrollo práctico: Para la realización de cada uno de estos módulos se siguieron metodologías específicas que nos ayudaron a conseguir el resultado deseado o lo más aproximado al resultado ideal. Para el caso de la UART todo se hizo a 115200 bps, se hizo el recibimiento de datos muestreando justo a la mitad de cada dato para asegurar el recibir el dato correcto y en la transmisión se calculó el dato de paridad y se añadió a el dato a enviar (cabe mencionar que nuestro envió cuenta con un bug a la hora de mostrar los datos en una terminal).

Para poder pasar los datos que se reciben atraves de la UART a nuestros procesadores estos tienen que pasar de un dominio de reloj a otro, para evitar problemas de meta estabilidad y que se respeten los tiempos de hold y de setup se usó una FIFO circular implementada con una RAM de dos puertos y dos relojes y un pequeño sistema para facilitar el cruce de dominios de reloj.

Para la realización de los procesadores se uso como base la teoría de arreglos sistólicos, en base a eso se hizo un diseño que tomara en cuenta el como se tenían que propagar los resultados de las operaciones para cablear las conexiones entre los procesadores y asi por medio de multiplexores y máquinas de estado emular por hardware la operación de matriz por vector con 4 procesadores en paralelo aunque esta quedo limitada hasta la una operación de 4x4.

Conclusiones.

A manera de cierre podemos decir que esta práctica fue bastante retadora pese a que no pudimos lograr los objetivos propuestos al cien por ciento los avances que se dieron en cuestiones respecto a cómo pensar en base hardware y como usar un lenguajes de descripción de hardware fueron notables, las soluciones que se fueron probando e implementando a lo largo de la practica fueron más avanzadas de lo que se había intentado en prácticas pasadas. Pese a esto la versión final no es lo deseado ya que contiene fallas que pudieron haber sido reparadas con más atención y práctica.

Videos:

<https://youtu.be/pJJE508e3CI> UART TX explicacion de errores

<https://youtu.be/I6QYhuNdeEs> Clock Domain Crossing

<https://youtu.be/SIs52Vi-hd4> Procesadores

<https://youtu.be/pPZnSpjcBUs> OP 2X2

<https://youtu.be/4DSVK-nS5Ig> Operacion 3X3

<https://youtu.be/TT3oJQBFI5w> UART en codigo

<https://youtu.be/3Erqp3TmU4M> Procesadores en codigo

do runfifo.do CDC example

do runp.do processors

do runtx.do uart transfer