

IE-0523 Circuitos Digitales II

Avance I Proyecto 1:
Diseño de la capa PHY de la interfaz PCIe

Brown Ramírez, Timna Belinda B61254
Esquivel Molina, Brandon B52571
Gutiérrez Muñoz, Michelle B43195
timna.brown@ucr.ac.cr
brandon.esquivel@ucr.ac.cr
michelle.gutierrezmunoz@ucr.ac.cr

I-2020

1. Resumen

Se llevó a cabo el diseño de la capa PHY de la interfaz PCIe usando el lenguaje de descripción de hardware verilog. Seguido de la descripción del plan de trabajo empleado, se encuentra una descripción detallada acerca de cada uno de los módulos que conforman las capas del transmisor y receptor, además de las conexiones entre estos mismos, se ejemplifica además el comportamiento deseado para algunos de los módulos. Después se describen de forma breve las pruebas más relevantes hechas para poder comprobar el funcionamiento del diseño implementado junto con una breve descripción del método utilizado para poder correr la simulación. Finalmente, se muestran los resultados obtenidos más relevantes del proyecto, haciendo uso de la herramienta gtkwave, seguido de esto, se concluyen con algunas conclusiones y recomendaciones, dentro de las cuáles se encuentran primeramente que se logró obtener el comportamiento deseado para cada uno de los módulos que conforman las capas phy_tx y phy_rx, además de que el principal desafío para la unificación de estas capas fue la sincronización de los módulos correspondientes.

2. Plan de trabajo

Las tareas se distribuyeron para aprovechar las capacidades de cada integrante. Las asignaciones planteadas inicialmente se muestran a continuación, sin embargo, los integrantes se involucraron en otras partes además de las asignadas.

- Avance 1 : Lógica de Mux L1 y L2 (Brandon), Recirculación, Clocks, Checkers y flip-flops (Belinda), Lógica de Demux L1 y L2 (Michelle)
- Avance 2: Serie a Paralelo COM/IDLE (Brandon), Paralelo a Serie COM/IDLE (Michelle), avance de módulos tx y rx (Belinda)
- Avance 3: Módulo phy_tx y Módulo phy_rx (Belinda), Pruebas finales (Brandon)
- Reporte y Presentación: Todos

3. Descripción Arquitectónica

A continuación se muestra un diagrama que representa la construcción arquitectónica del proyecto, desde los módulos individuales con tareas específicas hasta las capas superiores conformadas por los mismos y sus respectivas conexiones.

Detalles del PHY para el proyecto #1

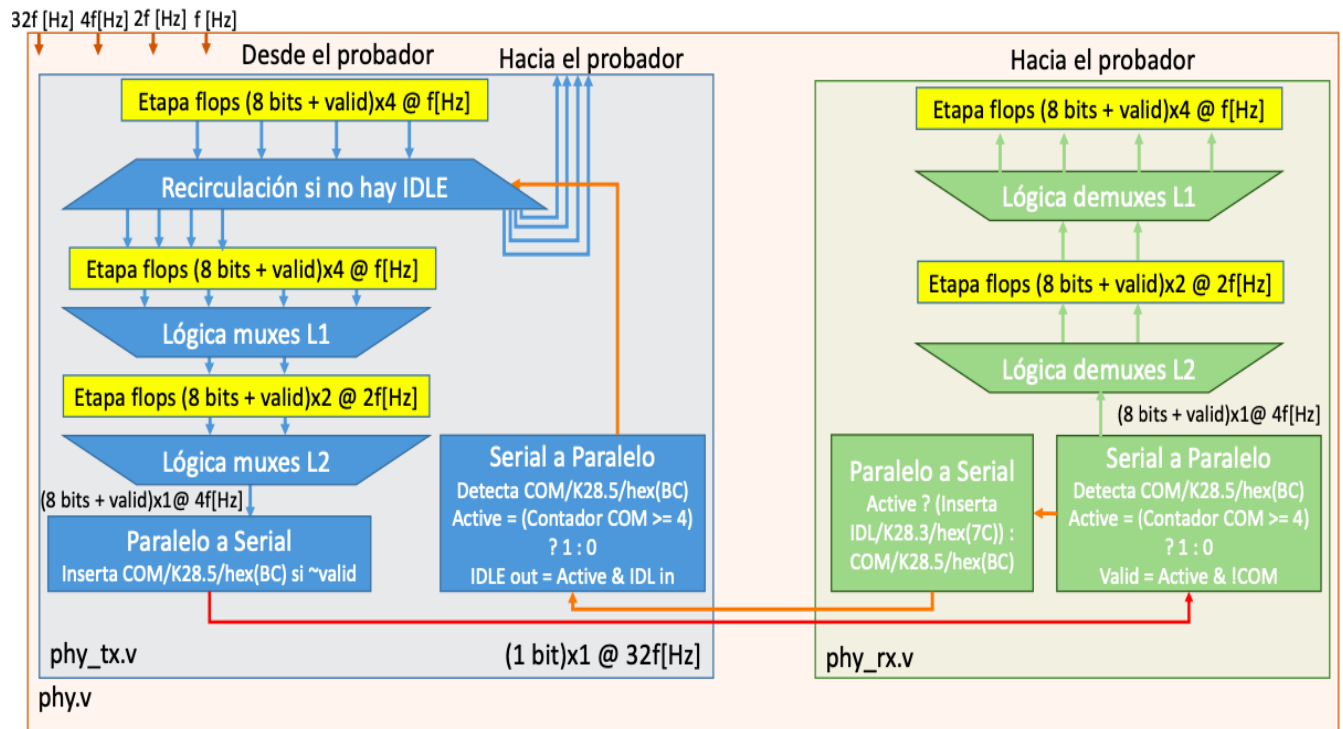


Figura 1: Diagrama que muestra la organización y construcción del proyecto

Ahora, se describirá cada uno de los módulos y capas mostrados en el diagrama.

3.1. Módulo Recirculación

Este módulo recibe 4 señales de datos y su respectivo valid, la señal de active procedente del módulo serial a paralelo, tiene 8 salidas de datos, 4 para la etapa de mux y 4 para el probador, cada uno con su respectivo valid. La función de este módulo consiste en pasar sus señales de entrada a sus salidas para el mux (en caso de que el active esté en alto) y por ende poner 0 en sus salidas para el probador, en caso contrario (active en bajo) hace lo opuesto, manda las entradas al probador y al mux envía salidas en 0. En otras palabras, recibe la señal de active y dependiendo del estado de la misma, recircula los datos al probador o los deja pasar a la etapa de mux.

3.2. Módulos de Muxes

A la etapa de muxes entran 4 señales y sale 1 señal a una frecuencia de 4. Primero se tiene una lógica de muxes L1 que corresponde a un mux 4:2, las 4 entradas son sincronizadas a una frecuencia f por una primera etapa de flip-flops, las dos salidas se sincronizan a una frecuencia de $2f$, seguido de esto, van a una lógica de muxes L2 que corresponde a un mux 2:1, cuya salida se sincroniza con una etapa de flip-flops que la sincroniza a una frecuencia de $4f$. Esto se implementó, creando y pobando por aparte el mux 4:2 y 2:1, luego de esto, se implementó el mux 4:1, haciendo uso de los anteriores e implementando también las etapas de flip-flops correspondientes. En la siguiente Figura se muestra el comportamiento esperado de las señales al pasar por cada etapa de lógica de muxes:

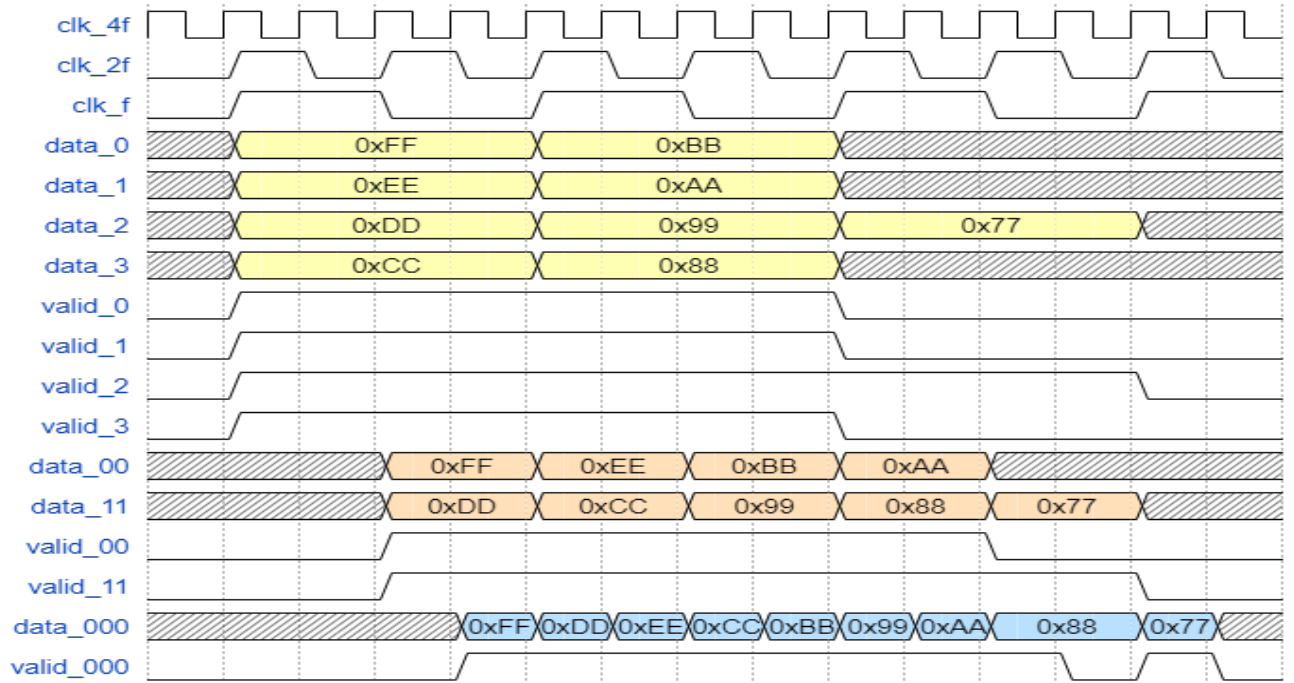


Figura 2: Digrama de señales que muestran el comportamiento esperado para la etapa de multiplexación

3.3. Módulos de Demuxes

De forma inversa a los muxes, los demuxes revierten el proceso, a estos entra 1 señal y salen 4 señales. Primero se tiene una lógica de demuxes L2, que corresponde a un demux 1:2, cuya entrada viene sincronizada a una frecuencia de 4f, las 2 salidas se sincronizan con una etapa de flip-flops a una frecuencia de 2f y luego van a la lógica de demuxes L1, que corresponde a un demux 2:4, cuyas 4 salidas se sincronizan a una frecuencia de f con otra etapa de flip-flops. Esto se implementó, creando y probando por aparte un demux 1:2 y 2:4, luego de esto, se implementó el demux 1:4, haciendo uso de los anteriores e implementando las etapas de flip-flops correspondientes.

3.4. Módulos de Flip-Flops, Clocks y Checkers

Los flip-flops son módulos que permiten sincronizar las señales a la frecuencia deseada, básicamente sus salidas son sus entradas (solo que ya sincronizadas). El módulo de clocks, por su parte, consiste de una lógica que hace posible obtener salidas que corresponden a clocks a diferentes frecuencias, de 1 Hz, 2 Hz, 4 Hz y 32 Hz, las que se necesitan para los demás módulos. Por último, los checkers tienen como función poder comparar la similitud en el comportamiento que debe existir entre los módulos conductuales y sus análogos estructurales, sintetizados por yosys. Esto se hace a través de señales que pasaran a alto en caso de que las salidas de un módulo sean diferentes al otro, de esta forma se puede verificar cuando estas diferencias podrían deberse a atrasos o a un comportamiento diferente.

3.5. Módulos de Paralelo-Serie

El módulo tiene una salida de 1 bit y recibe una señal de entrada compuesta por un bus de 8 bits, una señal de valid de 1 bit para esta entrada y adicionalmente al igual que los demás módulos tiene como entrada reset y un clock, en este caso a una frecuencia de 32, su función principal es la de "descomponer" la señal recibida de 8 bits y pasarla en su salida bit por bit, debido a esto es necesario aumentar la frecuencia a la salida y poder mantener el flujo de datos. En el caso del módulo paralelo-serie con idle (en el rx), en lugar de "descomponer" una entrada de datos de 8 bits, este recibe únicamente una señal de active, si esta se encuentra

en alto, entonces el módulo envía una señal correspondiente a 7C bit por bit, en caso contrario, envía una señal correspondiente a BC bit pot bit. Esta señal de salida servirá como indicador para recirculación, acerca de cuando dejar los datos correspondientes. Siempre que la salida corresponda a BC, será considerada como inválida por los módulos que la reciban. A continuación, se muestra el comportamiento esperado para las señales del módulo.

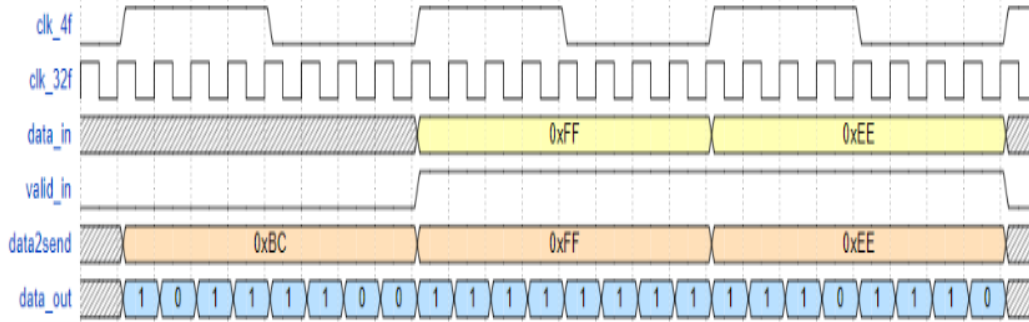


Figura 3: Digrama de señales que muestran el comportamiento esperado para un módulo paralelo-serie

3.6. Módulos de Serie-Paralelo

En este caso, por el contrario, se recibe una entrada de 1 bit a una frecuencia de 32 (además de reset y clocks a diferentes frecuencias) y su salida será una señal de 8 bits a una frecuencia de 4, para que la salida corresponda a una composición de 8 entradas. Ambos módulos de serie-paralelo (tanto en el tx como rx), se activan después de recibir 4 señales BC (se controla por medio del contador COM). En el caso del serie-paralelo en el rx, este tiene una señal de valid que dependerá del estado de active y COM, para asegurar que se están recibiendo datos válidos, la salida de este módulo irá a etapa de demuxes. Para el serie-paralelo en el tx, recibe una señal no válida de BC o una de IDLE (proveniente del módulo paralelo a serie del rx) y cuando este se encuentra activo y recibe la señal de IDLE la envía al módulo de recirculación. A continuación, se muestra el comportamiento esperado para las señales del módulo.

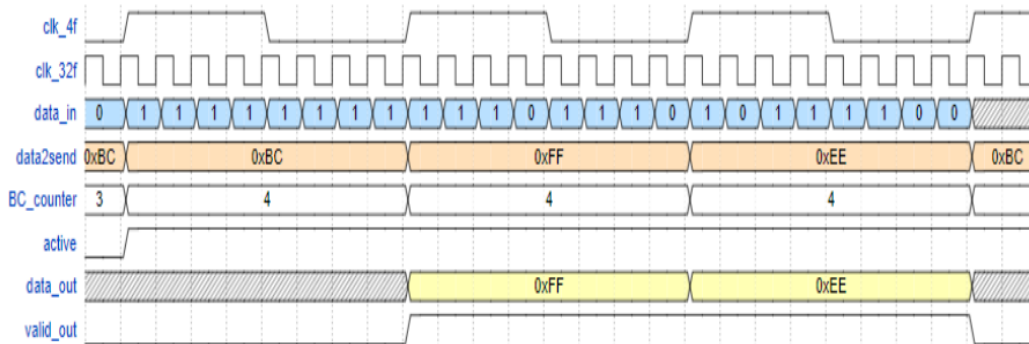


Figura 4: Digrama de señales que muestran el comportamiento esperado para un módulo serie-paralelo

3.7. Módulo Transmisor TX (phy_tx.v)

Este módulo debe incluir los módulos de recirculación, etapas de muxes y sus respectivos flip-flops, paralelo a serie COM y serie a paralelo idle. Las entradas (y su respectivo valid) vienen del probador, se sincronizan con una primera etapa de flip-flops luego se conectan al módulo de recirculación, dependiendo del active (otra entrada del módulo), vuelven al probador (salidas del módulo) o continúan hacia la etapa de multiplexación, la única salida de esta etapa es una salida del módulo.

3.8. Módulo Receptor RX (phy_rx.v)

Este módulo debe incluir los módulos de serie a paralelo COM, paralelo a serial idle, etapas de demuxes y sus respectivos flip-flops. La entrada del módulo, es de un bit y viene del módulo tx, primero entra al módulo serie a paralelo COM, seguido de esto, este módulo pasa la señal de active al paralelo a serie idle, cuya salida es una salida del módulo (para el tx), el módulo serie a paralelo COM también envía una señal de datos de 8 bits a la etapa de demultiplexación, al final se obtiene 4 salidas con su respectivo valid, estas son salidas del módulo.

3.9. Módulo Final (phy.v)

Esta capa final Phy, está compuesta por los módulos TX y RX. Sus entradas son las 4 señales de 8 bits y su respectivo valid (que entran primeramente al módulo TX) y sus salidas corresponden a 4 señales de 8 bits y su valid, que vienen del módulo RX y otras 4 salidas que vienen de la recirculación. Sus conexiones internas son de la salida del módulo paralelo a serie del TX al serie a paralelo del RX (flujo de datos), y de la salida del paralelo a serie del RX al serie a paralelo del TX (señal de idle), ambas señales son de 1 bit y tienen frecuencia de 32 Hz.

4. Plan de Pruebas

Pruebas en Muxes y Demuxes

- 1 Verificación del funcionamiento de los valid : Después de usar el valid aparte de las señales (al inicio se pusieron como el bit 0 de las señales, siendo estas entonces de 9 bits), se varió el estímulo de las señales de valid desde el probador para luego verificar que el comportamiento deseado ocurriera cuando los valid están en alto.
- 2 Comportamiento general de los mux/demux en las capas intermedias: Se hicieron pruebas de más tiempo, y cambiando más los estímulos correspondientes, para verificar así que se obtuviera el comportamiento deseado.
- 3 Comportamiento general del mux 4:1 :
- 4 Comportamiento general del demux 1:4 sin clks : Se cambiaron los estímulos en el probador al igual que antes, los resultados presentaron problemas, dos salidas indeterminadas.
- 5 Comportamiento general del demux 1:4 sin clks : Igual que la prueba anterior pero se dejó el reset en 0 por mucho más tiempo, se arregló el problema.
- 6 Comportamiento general del demux 1:4 con clks: Se probó con los mismos estímulos que antes, pero el problema solucionado reapareció.
- 7 Comportamiento general del demux 1:4 (v2) con clks: Igual que prueba anterior, persistió el problema (esta vez se modificó la estructura interna del demux).
- 8 Comportamiento general del demux 1:4 (v3) con clks: Se cambiaron los valores de los estímulos, se verificó el comportamiento cuando el valid está en alto y bajo, también que el reset dure lo suficiente, demux funcional (esta es otra la versión 3).
- 9 Comportamiento general del mux 4:1 y demux 1:4 con clks: Se ingresaron los mismos estímulos que se muestran en las imágenes de guía tanto para el demux y mux, y se verificó que se obtenía el comportamiento deseado, prueba exitosa.

Pruebas de módulos serie-paralelo y paralelo-serie

- 1 Comportamiento general del paralelo-serie COM: Se cambiaron los estímulos desde el probador, igual a los descritos en las imágenes de guía, se comprobó el comportamiento deseado pero con un retardo.

- 2 Comportamiento general del paralelo-serie COM (v2): Se corrigió un problema en el código que generaba un buffer, se repitió la prueba, se eliminó el retardo.
- 3 Comportamiento general del paralelo-serie idle: Se cambió el estímulo desde el probador, para que el active estuviera activo y no activo en diferentes momentos, además de que el valid no siempre estaba activo, se verificó el comportamiento deseado.
- 4 Comportamiento general del serial-paralelo COM: Desde el probador se enviaron 4 señales BC para corroborar que el módulo se activa, luego se enviaron diferentes señales, luego se envía de nuevo un dato BC para verificar el comportamiento de la señal de valid_out.
- 5 Comportamiento general del serial-paralelo idle: Se hizo un probador similar al del módulo COM, se probó además, con la señal de idle.

Pruebas en los módulos phy

- 1 Comportamiento general de los módulos: Se mantiene el reset por varios ciclos en 0, para asegurar que los flip-flops se reseten apropiadamente, igual que antes.
- 2 Comportamiento al unir los módulos: Se cambian de forma apropiada los estímulos en el probador, se obtienen problemas de sincronización de módulos.

Comparación de módulos

Adicionalmente, se hicieron pruebas empleando tanto los módulos conductuales, como sus análogos estructurales (sintetizados con yosys), por medio de los checkers hechos, para poder verificar que el comportamiento era el mismo en ambos.

5. Instrucciones para utilización de la Simulación

Se debe hacer uso del archivo del makefile (se encuentra en el git), el cuál contiene todos los comandos necesarios para poder correr toda la simulación completa del proyecto.

6. Resultados Obtenidos

A continuación se muestran los resultados de las etapas de multiplexación, demultiplexación, paralelo-serie COM y serie paralelo idle (de comportamiento similar a sus módulos análogos) usando el programa gtkwave para poder observar el comportamiento de las señales. *Módulos de Multiplexación*

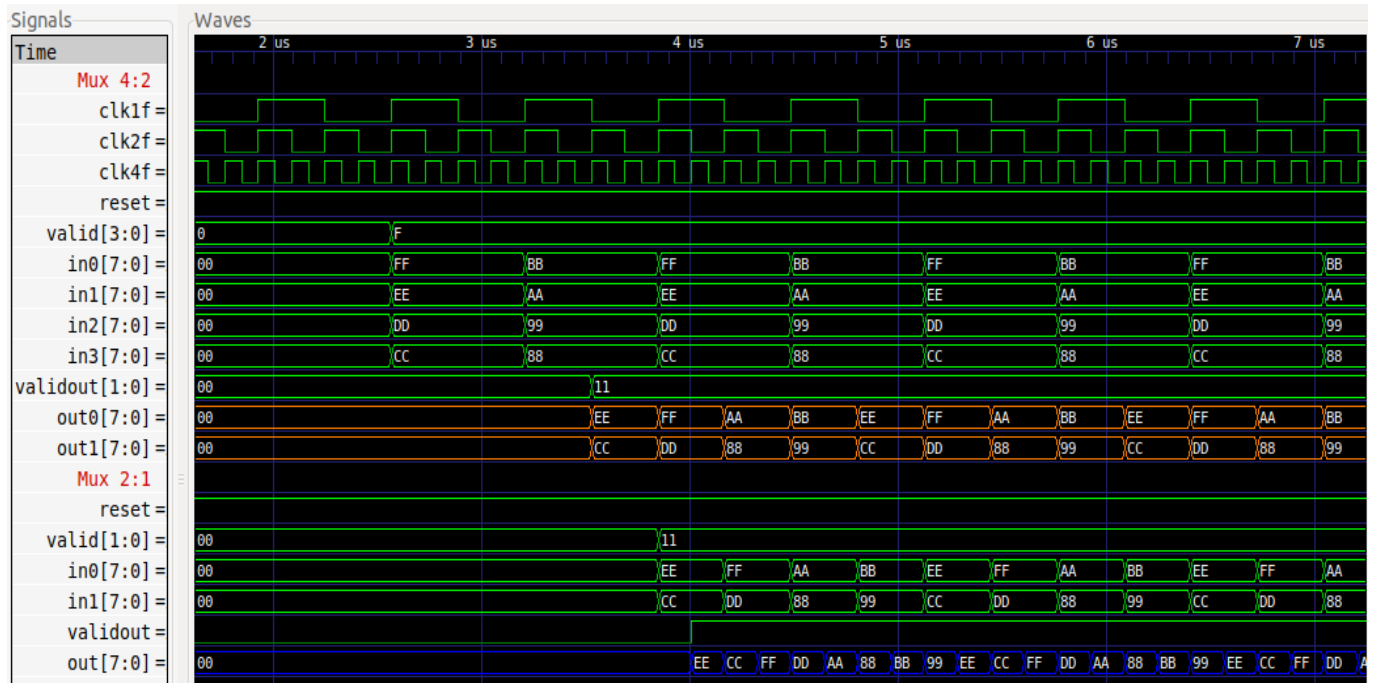


Figura 5: Comportamiento obtenido de los módulos del multiplexación

Módulo de paralelo-serie COM

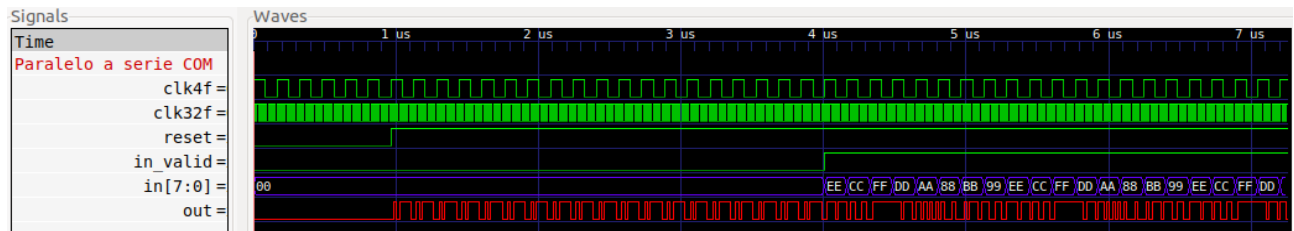


Figura 6: Comportamiento obtenido del módulo paralelo-serie COM

Módulo de serie-paralelo idle

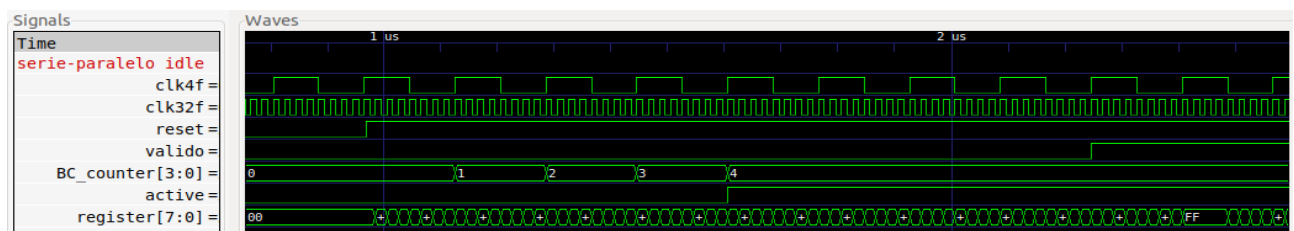


Figura 7: Comportamiento obtenido del módulo serie-paralelo idle

Módulo phy_tx

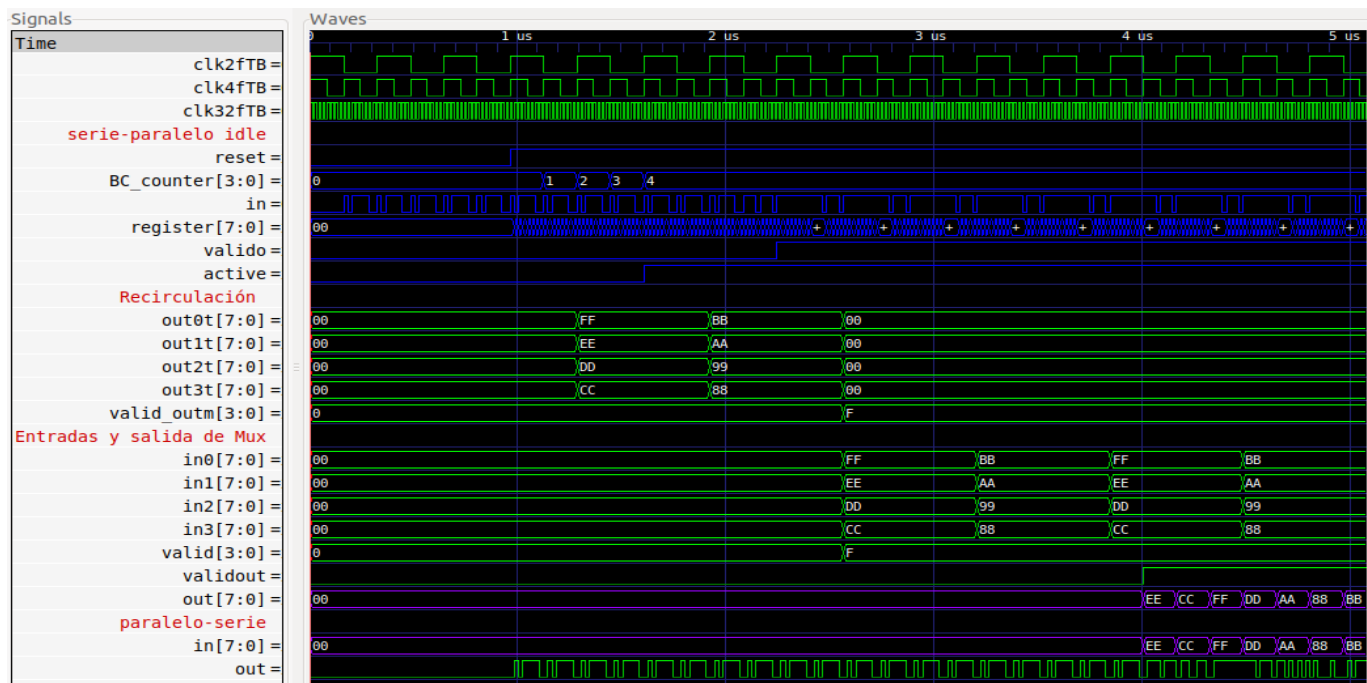


Figura 8: Comportamiento general obtenido para el módulo transmisor `phy_tx`

Módulos de Demultiplexación

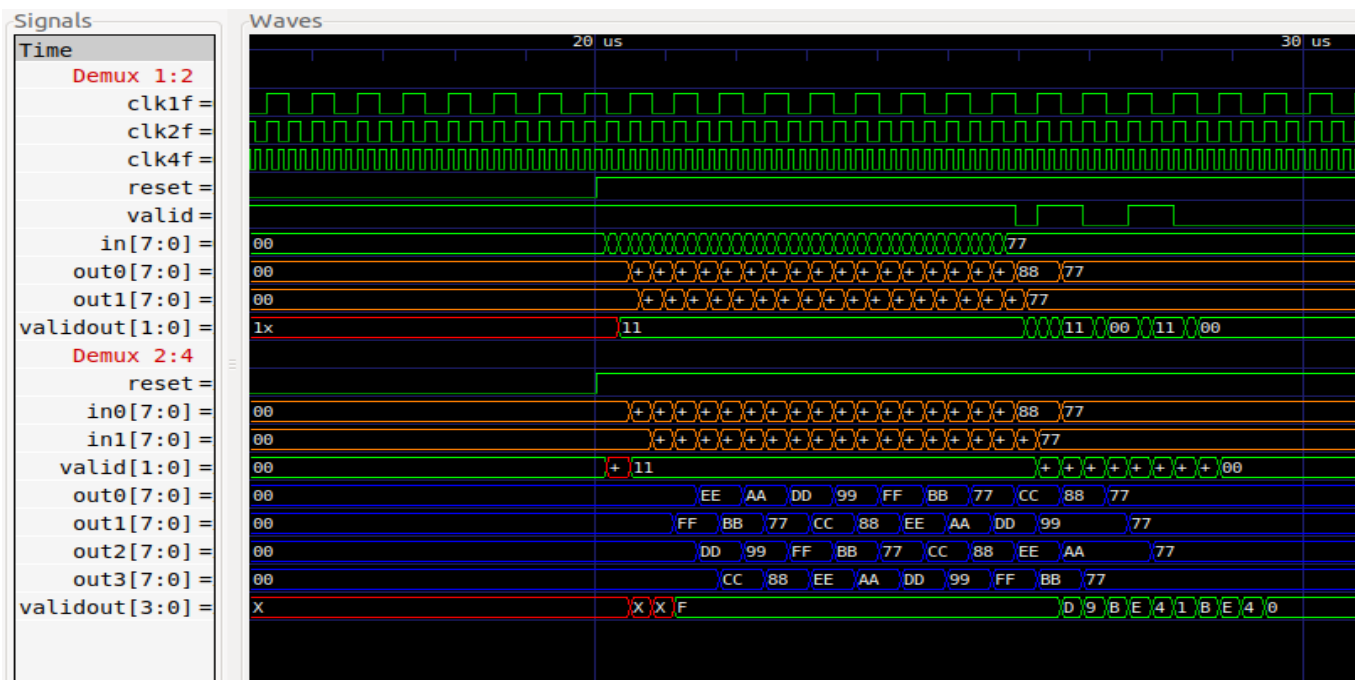


Figura 9: Comportamiento obtenido de los módulos del demultiplexación

Módulo `phy_rx`

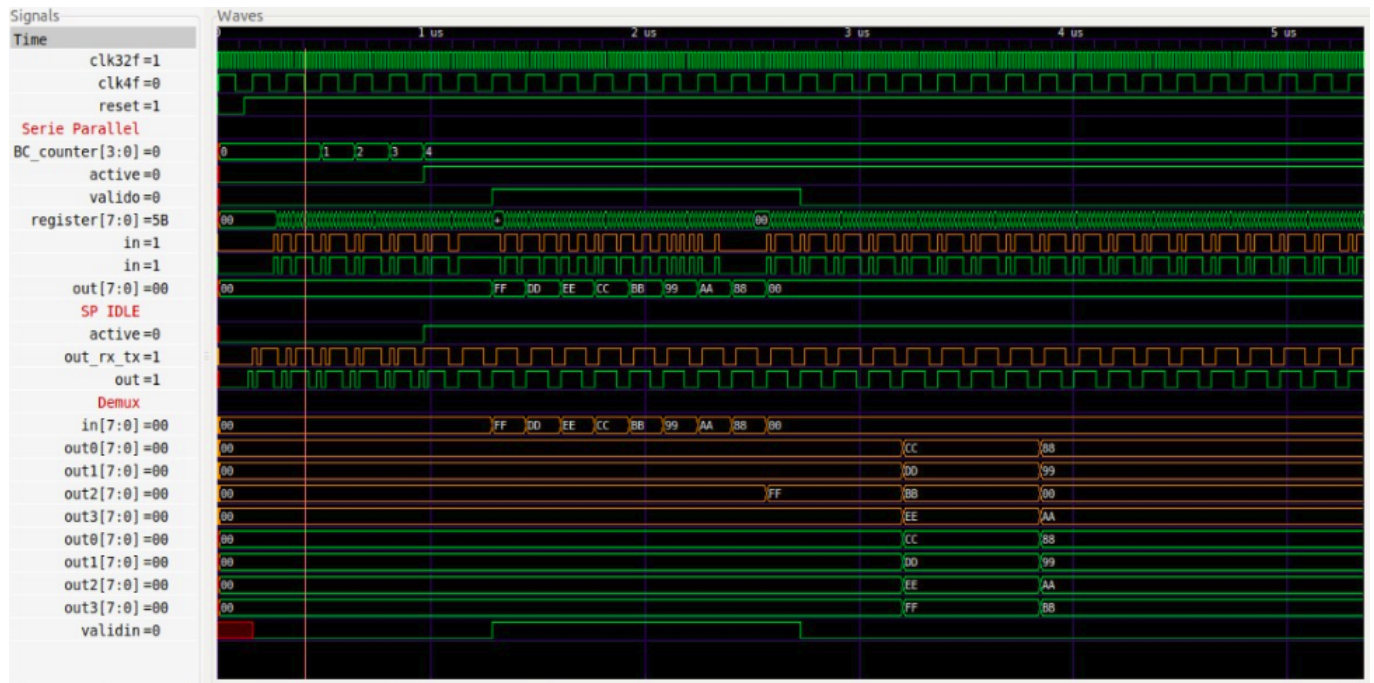


Figura 10: Comportamiento general obtenido para el módulo receptor phy_rx

7. Conclusiones y Recomendaciones

- Se obtuvieron los resultados esperados para todos los módulos que conforman las capas phy_tx y phy_rx.
- Se encontraron bastantes desafíos al implementar la herramienta de síntesis de yosys, ya que los resultados de la misma a veces no se comportan de la forma esperada.
- Se encontró como ventaja el uso de la plataforma de git lab, ya que permite llevar una noción global del avance del trabajo, además de poder así compartir y modificar los archivos entre los integrantes.
- Se cambió el diseño original de la etapa de demultiplexación (al inicio el demux 1:4 se construyó a partir de tres demux 1:2), la versión final cuenta con un demux 1:2 y otro demux 2:4.
- Se aprendió muchísimo más acerca del valor y utilidad de los módulos de tester y como estos son claves para corroborar el funcionamiento del diseño planteado.
- Se encontraron dificultades para manejar la sincronización de los diferentes módulos, especialmente en los sintetizados, donde existieron complicaciones con los reset.
- Se recomienda una mejor planificación del tiempo para proyectos futuros.
- Se resalta la importancia de la comunicación constante entre los integrantes del proyecto, para obtener mejores resultados.