

## Proyecto #2

### Capa de transacción PCIe adaptada y lógica de conmutación

#### Avances de trabajo

- Avance 1: 19 de junio, 17:00
- Avance 2: 26 de junio, 17:00
- Avance 3: 3 de julio, 17:00
- Avance 4: 6 de julio, 08:00
- Presentación: 10 de julio, 17:00

#### Investigación

Realice una breve investigación acerca de:

1. QoS (Calidad de Servicio)
2. Arbitraje en sistemas digitales
3. Priority Flow Control
4. ¿Cómo se relacionan los créditos con Flow Control?

#### Especificaciones

Las siguientes son las características básicas del diseño solicitado:

1. Las características de la capa de transacción para una interfaz PCIe y/o USB son definidas por un estándar. La arquitectura del proyecto se puede observar en la presentación QoS\_TC\_VC\_switch.pdf en Mediación Virtual.
2. El diseño del DUT se hará en el lenguaje Verilog y deberá incluir una descripción **conductual** que se usará como verificador automático y una descripción **estructural**. La descripción estructural se generará con el sintetizador YOSYS. Se debe comprobar la equivalencia de las descripciones de cada bloque por medio de una prueba.
3. Se debe tener un **Plan de Pruebas** definido para establecer que el diseño es 100% funcional. El plan de pruebas se debe validar con el profesor en la clase siguiente a la entrega del enunciado.
4. El grupo de tres estudiantes trabajará en equipo para completar esta asignación. Las características del equipo de trabajo serán las siguientes.
  - a) Entre los integrantes del equipo de trabajo se escogerá una persona para que sea el líder. Esta persona será responsable de convocar a los otros integrantes del equipo a reuniones y llevará una **bitácora** de los acuerdos, y avances que haga el equipo.
  - b) Entre los integrantes del equipo se compartirán las destrezas que cada uno de los integrantes tiene. De esta forma se pueden asignar roles y ciertas tareas. Dentro de las destrezas podrían estar: facilidad para planear, facilidad para programar, facilidad para hacer presentaciones en público, facilidad para escribir reportes, etc. La idea es que el equipo pueda repartirse las tareas del proyecto tomando en cuenta las destrezas de cada uno.
  - c) Cada integrante del equipo debe mantener informado al resto del equipo de los

avances que ha hecho en las tareas que está haciendo y el tiempo que ha tomado hacerlas.

- d) Como equipo de trabajo, su primera tarea será evaluar las propuestas de diseño de todos los integrantes y seleccionar una que se utilizará para ejecutar el proyecto. Esta propuesta se presentará en la primera presentación de avance que se hace en la clase.
5. Diseño del FIFO:
    - a) Construya o busque en Internet una memoria con un puerto de entrada y un puerto de salida (se debe poder leer y escribir al mismo tiempo). Sintetice la memoria utilizando Yosys y diseñe una prueba funcional que verifique el funcionamiento correcto de ambos modelos. Agregue una captura con ambas descripciones funcionando al reporte.
    - b) Construya un FIFO utilizando la memoria anterior. El FIFO debe comportarse de acuerdo con la presentación QoS\_TC\_VC.pdf, incluyendo las señales de Flow Control. Sintetice y genere una prueba. Agregue una captura con ambas descripciones funcionando al reporte.
  6. Diseño de lógica miscelánea:
    - a) Diseño de lógica para hacer “push” y “pop” de cada FIFO, además de los muxes y demuxes necesarios para el flujo de datos.
  7. Diseño de una Máquina de Estados.
    - a) Construya una Máquina de Estados con codificación one-hot de acuerdo con la descripción de la diapositiva “FSM de Control” en la presentación QoS\_TC\_VC.pdf. Sintetice y genere una prueba. Agregue una captura con ambas descripciones funcionando al reporte.
  8. Diseñe la lógica necesaria para interconectar los bloques anteriores de acuerdo a la Arquitectura de la presentación QoS\_TC\_VC.pdf. No olvide la lógica de registros de init y umbrales ni las consideraciones descritas en la presentación QoS\_TC\_VC.pdf. Sintetice y genere una prueba. Agregue una captura con ambas descripciones funcionando al reporte.
  9. Todas las pruebas deben tener un makefile que permita sintetizar, simular, visualizar y limpiar de forma automática.
  10. El proyecto deberá entregarse en un repositorio de git. La entrega de los avances del 7, 14 y 21 de noviembre será en forma presencial. Se hará un “git pull” del repositorio del equipo de trabajo y se correrá limpieza, síntesis, simulación y visualización de resultados en la computadora del profesor.

## Evaluación del Proyecto

- |  |               |
|--|---------------|
| 1. Plan de pruebas (2.5%)                                | (19 de junio) |
| 2. Bitácora (2.5%)                                       | (6 de julio)  |
| 3. Funcionamiento del diseño (85%)                       |               |
| a. Memoria (5%)  | (19 de junio) |
| i. Descripción conductual funcional (verificada) (2%)    |               |
| ii. Descripción estructural funcional (verificada) (2%)  |               |
| iii. Prueba con ambas descripciones funcionando (1%)     |               |
| b. FIFO (10%)  | (26 de junio) |
| i. Descripción conductual funcional (verificada) (4%)    |               |
| ii. Descripción estructural funcional (verificada) (4%)  |               |
| iii. Prueba con ambas descripciones funcionando (2%)     |               |
| c. Máquina de Estados (10%)                              | (19 de junio) |
| i. Descripción conductual funcional (verificada) (4%)    |               |
| ii. Descripción estructural funcional (verificada) (4%)  |               |
| iii. Prueba con ambas descripciones funcionando (2%)     |               |
| d. Lógica miscelánea e interconexión completa (60%)      | (3 de julio)  |
| i. Descripción conductual funcional (verificada) (24%)   |               |
| ii. Descripción estructural funcional (verificada) (24%) |               |
| iii. Prueba con ambas descripciones funcionando (12%)    |               |
| 4. Reporte del Proyecto (5%)                             | (6 de julio)  |
| 5. Presentación del Proyecto (5%)                        | (10 de julio) |