

# Diseño de microprocesadores

## Práctica 1

Realizar la implementación del procesador MIPS multi-ciclo visto en clase. Este procesador debe ser capaz de ejecutar el programa que implementa el factorial de  $n$ . El valor de  $n$  se transmitirá a través de una terminal serial, se recomienda el uso de docklight, pero puede ser Teraterm o PuTTY.  $n$  será recibida por el procesador por una UART la cual se tiene que integrar como periférico. Una vez procesado el factorial, el resultado será transmitido por la UART hacia la terminal en la PC. Note que la UART solo puede transmitir 8 bits a la vez, por lo cual para poder transmitir los 32 bits resultantes se tiene que hacer en paquetes de 8 bits, comenzando por los 8 bits más significativos.

## Entregables

Un reporte que contenga:

- Una captura de pantalla de la simulación de cada instrucción implementada donde señales los puntos clave de la ejecución en particular.
- Una tabla donde se muestre el CPI de cada una de las instrucciones implementadas.
- Grafica donde se muestre la relación  $n$  vs  $CPU_{time}$  para  $n = 0$  hasta  $n = 15$ .
- Resultados de síntesis en términos de logic elements (LEs) y la frecuencia máxima de operación del diseño.
- Micro-arquitectura en Visio o programa equivalente.
- Captura de pantalla de las señales internas usando el Signal Tap para la instrucción LW
- Todas las fuentes deben ser debidamente comentadas.
- Código fuente en Moodle y un repositorio en git hub, agregar al repositorio a jlpe-iteso. Por cada instrucción agregada se espera al menos un commit.

## Evaluación

- El presentar el procesador funcionando de manera correcta en conjunto con la UART representa el 100% de la calificación.
- Si se presenta el procesador funcionando solo en simulación representa el 80 % de la calificación.
- En caso no completarse el procesador se asignará calificación proporcional al avance obtenido.
- Por cada semana de atraso se penalizará con 20% del valor de la práctica.

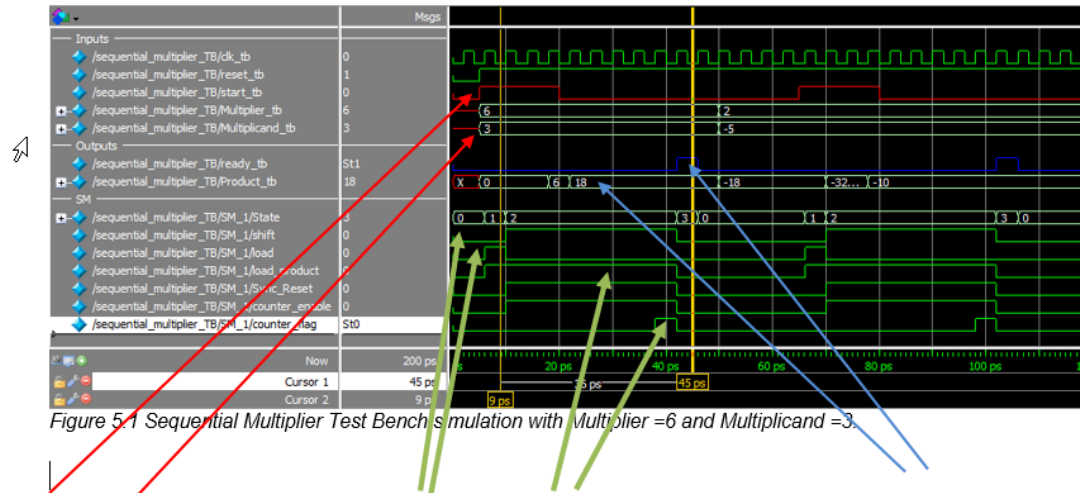
## Restricciones:

- Las direcciones memoria mostradas en la implementación/simulación tiene que coincidir con las mostradas en el MARS, es decir, el program counter comienza en la dirección 0x4000000 y la RAM en la dirección 0x10010000.
- La implementación del procesador deber ser a través de un diseño estructural, no se admitirán modelos comportamentales. Pero módulos constitutivos del procesador si puede implementarse de manera comportamental, ejemplo, la ALU.

## Recomendación:

Para la simulación se puede omitir el circuito generador de frecuencia.

## Formato para reportar simulaciones:



Multiplier and Multiplicand valid input data with the "start" signal activation.

States:  
IDLE=0  
LOAD=1  
SHIFT=2  
READY=3

"shift" and "load" signals for the shift registers.

"load\_product" and "Synch\_Reset" signals for the Accumulator.  
Note1.

"counter\_enable" and "counter\_flag" for the Counter block used by the FSM.

Product result of 18 (Valid until ready signal activation).

"ready" signal active indicating a valid "Product" result.