

9

Parcial 1 - Arquitectura de computadoras 2024  
Nombre: TOMÁS ACHAVAL BEZERRA

Ejercicio 1

a) Dibujar en el recuadro el circuito resultante de la síntesis del siguiente código en SystemVerilog:

```

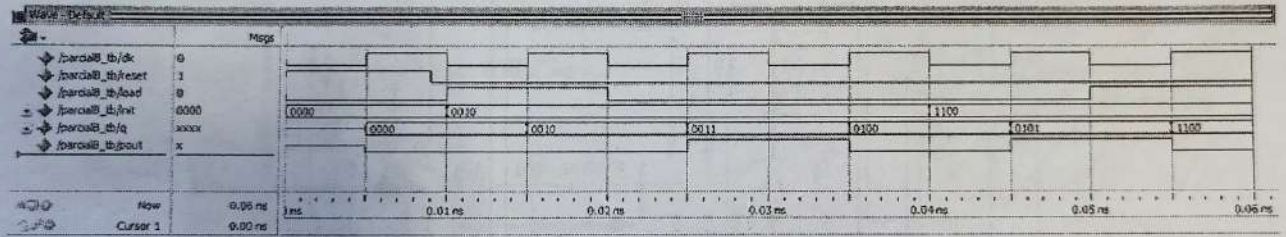
module parcial
  (input logic clk, reset, load,
   input logic [3:0] init,
   output logic [3:0] q,
   output logic pout);

  always_ff @(posedge clk)
    if (reset) q <= 4'b0;
    else if (load) q <= init;
    else q <= q + 4'b0001;
  assign pout = q[0];
endmodule

```

b) ¿Qué nombre le pondría a este módulo en lugar de "parcial", que sea más representativo del circuito descrito? \_\_\_\_\_

c) A continuación se describe en SystemVerilog el test bench del módulo parcial. Completar las líneas con los elementos faltantes para obtener las formas de onda de respuesta que se muestran en la figura.



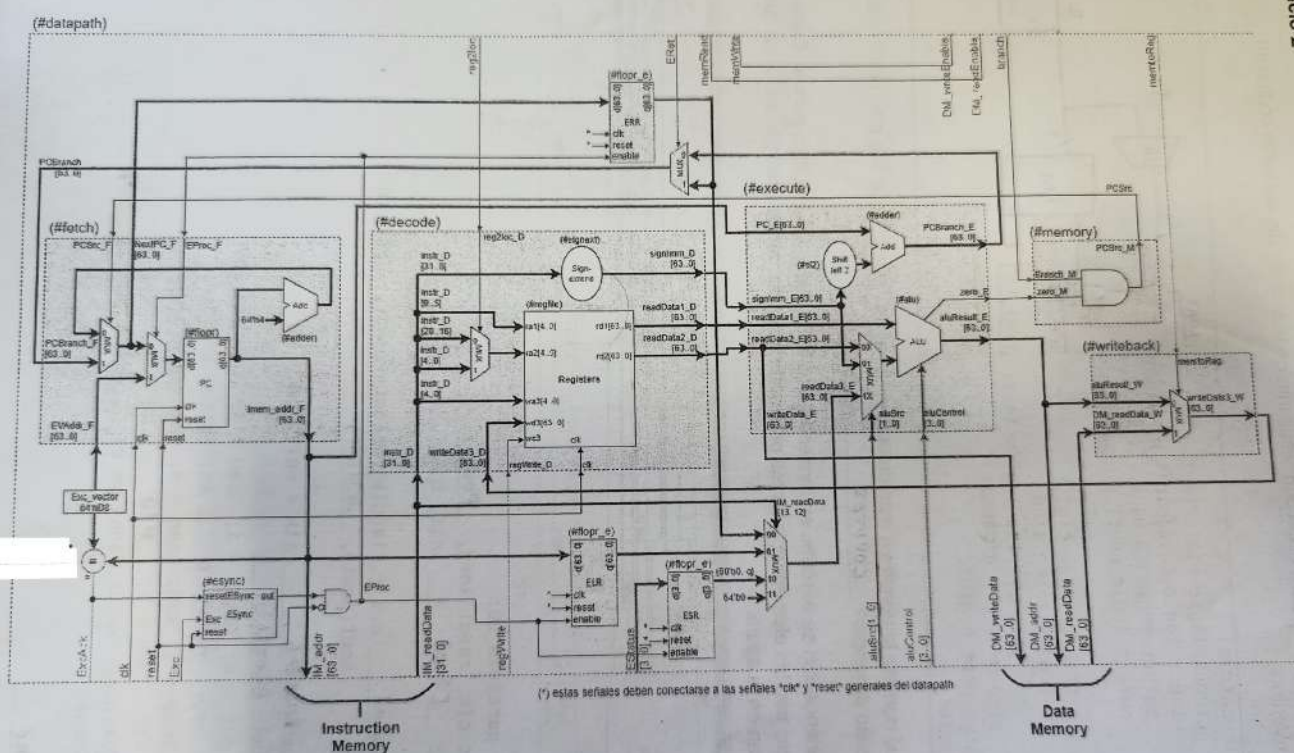
```

module parcialB_tb;
  logic clk, reset, load, _____;
  logic _____ init, q;

  _____ dut (.clk(clk), .reset(reset), .load(load),
               .pout(pout), _____, .q(q));

  always _____ clk = ~clk; _____
  initial _____
    clk = 0; reset = 1; load = 0; _____;
    #9; _____; #1; init = 4'b0010;
    load = 1; #10; load = 0; #20; _____;
    #10; load = 1; _____;
    $display("Valor final de q: _____", _____);
    $stop;
end
_____

```



(\*) estas señales deben conectarse a las señales "clk" y "reset" generales del datapath

Nombre  
Se requiere el manejo de  
el sistema de  
(Extr...



45085745 A.

45085746 A.

### Parcial 1 - Arquitectura de computadoras 2024

Nombre: TOMÁS AGUIRRE BEZERRA

Se requiere analizar las modificaciones necesarias al procesador LEGv8 de un ciclo con manejo de EXCEPCIONES (Práctico 3) a fin de separar los vectores de excepciones de sistema (op code invalido para nuestro caso), del vector de interrupciones por periféricos (ExtIRQ). Los nuevos vectores son:

- OpCode Invalido = 0x00D8
- Interrupción de E/S externa (ExtIRQ) = 0x00F8

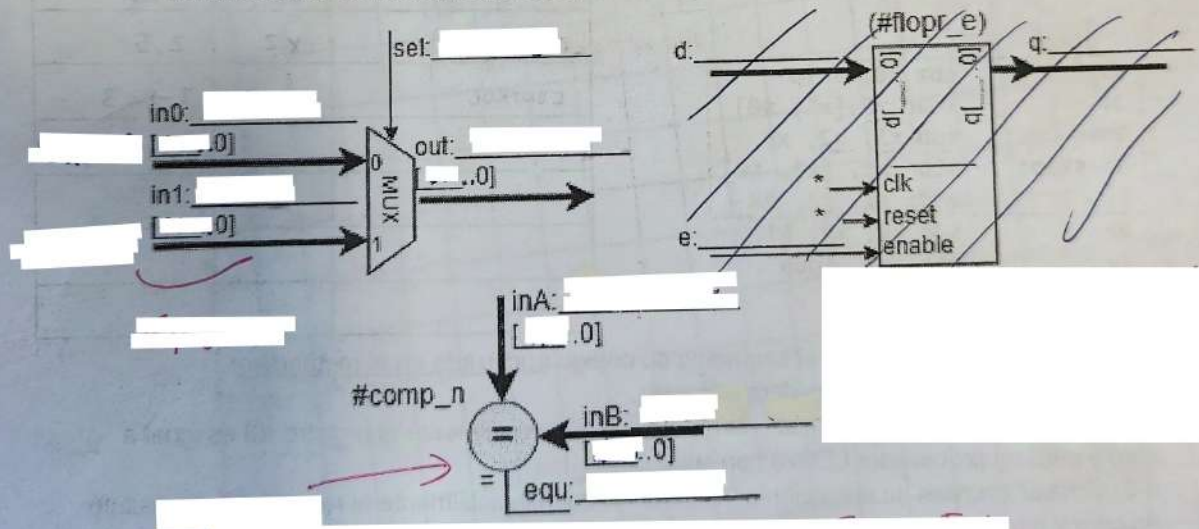
**NOTA:** La señal de respuesta *ExcAck* solo debe ponerse en "1" en caso de ejecutarse la ISR de una Interrupción de E/S externa. Caso contrario debe permanecer en "0".

Proponga una modificación al procesador y responda:

- a) Es necesario agregar señales de control extras?  
 NO..... Si.....
- | Nombre | #Origen | #Destino |
|--------|---------|----------|
| _____  | _____   | _____    |
| _____  | _____   | _____    |

- b) Enumere qué módulos existentes deben ser modificados (si los hubiera)
- 1 \_\_\_\_\_ 2 \_\_\_\_\_  
 3 \_\_\_\_\_ 4 \_\_\_\_\_

- c) Se requiere agregar alguno de éstos módulos? De ser afirmativo, especificar el nombre de las señales involucradas. Caso contrario, tachar el/los módulos.



- d) Suponga que el contenido de la dirección de memoria 0x00FC no contiene una instrucción válida. Que ocurre en nuestro nuevo procesador si ocurre una interrupción de E/S externa? (marcar la correcta)
- La ISR de interrupción de E/S externa se ejecuta normalmente
  - El procesador se queda en un bucle infinito sobre el vector 0x00F8
  - El procesador se queda en un bucle infinito sobre el vector 0x00D8
  - Ninguna de las anteriores



### Ejercicio 3

Asumiendo que las etapas individuales del pipeline de un procesador tienen las siguientes latencias:

	IF	ID	EX	WB
Latencia por etapa	100 ns	80 ns	130 ns	105 ns

a) Completar:

Tipo de procesador	Máxima frecuencia de reloj [Hz]	Latencia de una instrucción [ns]
Sin pipeline		
Con pipeline		

Asuma que quiero ejecutar un fragmento de código que tiene 2E6 instrucciones, las cuales no presentan hazards: ↘ 2000000

b) ¿Cuánto tiempo llevaría ejecutar dicho fragmento de código? \_\_\_\_\_

c) ¿Cuánto tiempo llevaría ejecutar el mismo fragmento si el procesador no tuviese pipeline? \_\_\_\_\_

### Ejercicio 4

Para la siguiente secuencia de código:

```

loop:
1>      cbz x0, skip
2>      ldur x2, [x1, #0]
3>      sub x3, x2, x0
4> skip: stur x3, [x1, #0]
5>      addi x1, x2, #8
6>      sub x9, x9, #1
7>      cbnz x9, loop
    
```

Tipo de dependencia	Registro	Instrucciones

a) Considerando únicamente el segmento de código encerrado en el rectángulo:

a-1) Completar la tabla con las dependencias.

a-2) Mostrar el orden de ejecución considerando que inicialmente el registro X0 es igual a cero y para un procesador LEGv8 con stall.

a-3) Mostrar el orden de ejecución considerando que el inicialmente el registro X0 es distinto de cero y para un procesador LEGv8 con forwarding stall.

b) Considerando el código completo, que inicialmente el registro x9 está inicializado en 1000, x0 es distinto de cero y para un procesador LEGv8 con forwarding stall:

b-1) ¿Cuántos ciclos de clock tomaría ejecutar este código? \_\_\_\_\_

b-2) ¿Cuántos ciclos de clock tomaría ejecutar este código en el microprocesador optimizado para saltos: \_\_\_\_\_

b-3) Calcular la ganancia de velocidad entre los puntos b-1 y b-2: \_\_\_\_\_

Respuestas puntos a-2 y a-3 en la siguiente hoja

