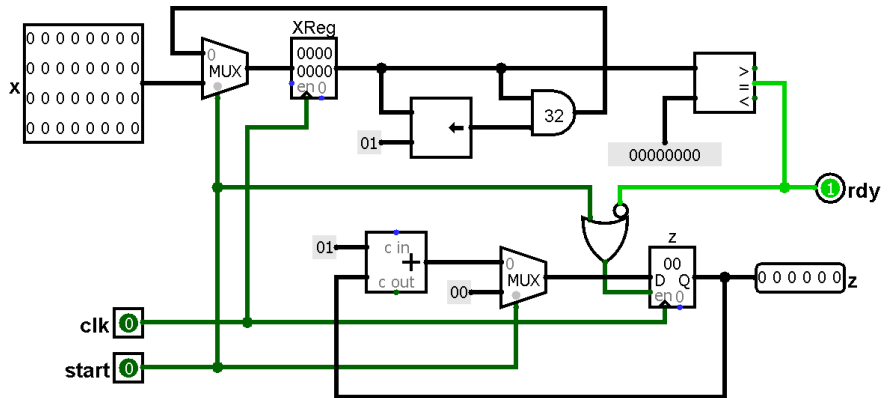


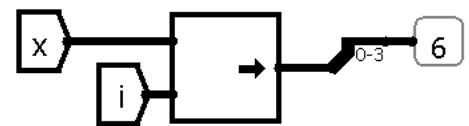
Pregunta 1

Escriba en system verilog el módulo de nombre *p1* equivalente al circuito de logisim de la derecha. Su módulo debe ser sintetizable a un circuito que ante las mismas entradas, produzca las mismas salidas que el circuito de la derecha.



Pregunta 2

El siguiente es un módulo en system verilog. Sintéticelo a un circuito equivalente en logisim. Para mejor claridad use túneles de logisim. *Ayuda:* La notación $x[i +: 4]$ es un *bit slice* y es equivalente al circuito de la derecha, es decir los 4 bits menos significativos del resultado de desplazar x en i bits hacia la derecha. Por ejemplo, si x es el hexadecimal 12345678 e i es 8, entonces el resultado es el valor 6.



```
module p2( input clk, start,
           input [31:0] x,
           output rdy, isp );

reg [4:0] i, j;

assign isp = x[i +: 4] == x[j +: 4]; // bit slice
assign rdy = i ≤ j || !isp;
```

```
always_ff @(negedge clk)
  if (start) begin
    i <= 28;
    j <= 0;
  end
  else if (!rdy) begin
    i <= i - 4;
    j <= j + 4;
  end

endmodule
```

Pregunta 3

Parte a.- En el circuito de la derecha se ha agregado memoria RAM a LRV32IM.

Conteste la siguientes preguntas:

- ¿De qué tamaño es el módulo de memoria RAM en KB?
- ¿En qué rango de direcciones se agregó? Expréselo como un intervalo como [inicio, fin[en hexadecimal.
- ¿Por qué no se conectaron las líneas A1 y A0 a la dirección de los módulos de memoria?
- Explique por qué esta interfaz de memoria es incorrecta.

Ayuda: $0x1000 \equiv 4 \text{ KB}$

Parte b.- Corrija esta interfaz de memoria.

