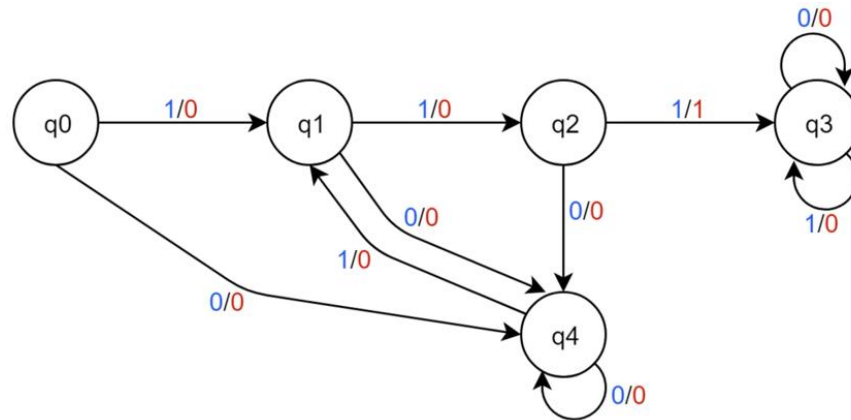
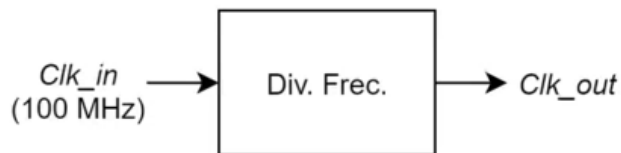


MAQUINA DE ESTADOS

Implementar la siguiente máquina de estados utilizando un divisor de frecuencia



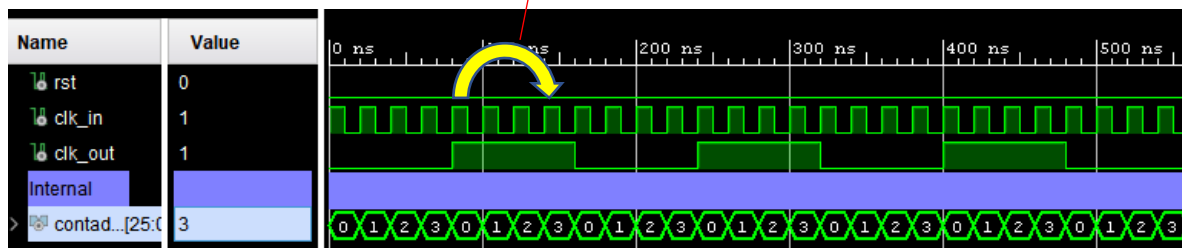
Primeramente, se realiza el divisor de frecuencia mediante el uso de un contador, en este caso se implementa el divisor de frecuencia tan solo de 4 pulsos a la entrada por uno de salida ya que se pretende observar su funcionamiento.



```

divisor: process (rst, clk_in)
begin
  if (rst = '1') then
    temp <= '0';
    contador <= (others => '0');
  elsif rising_edge(clk_in) then
    if (contador = 3) then
      temp <= not(temp);
      contador <= (others => '0');
    else
      contador <= contador + 1;
    end if;
  end if;
end process;
clk_out <= temp;
  
```

4



Posteriormente, se realizó la implementación de la maquina de estados la cual depende de la variable x para cambiar de estado, y como variable de salida tiene la variable z, la cual depende del estado inicial y de la variable x.

```

proceso_1: process (edo_presente, x)
begin
  case edo_presente is
    when q0 => z <= '0';
    if x = '1' then
      edo_futuro <= q1;
    else
      edo_futuro <= q4;
    end if;
    when q1 => z <= '0';
    if x = '1' then
      edo_futuro <= q2;
    else
      edo_futuro <= q4;
    end if;
    when q2 =>
    if x = '1' then
      edo_futuro <= q3;
      z <= '1';
    else
      edo_futuro <= q4;
      z <= '0';
    end if;
    when q3 => z <= '0';
    if x = '1' then
      edo_futuro <= q3;
    else
      edo_futuro <= q3;
    end if;
    when q4 => z <= '0';
    if x = '1' then
      edo_futuro <= q1;
    else
      edo_futuro <= q4;
    end if;
  end case;
end process proceso_1;

proceso_2: process (clk)
begin
  if (clk'event and clk='1') then
    edo_presente <= edo_futuro;
  end if;
end process proceso_2;

```

4

