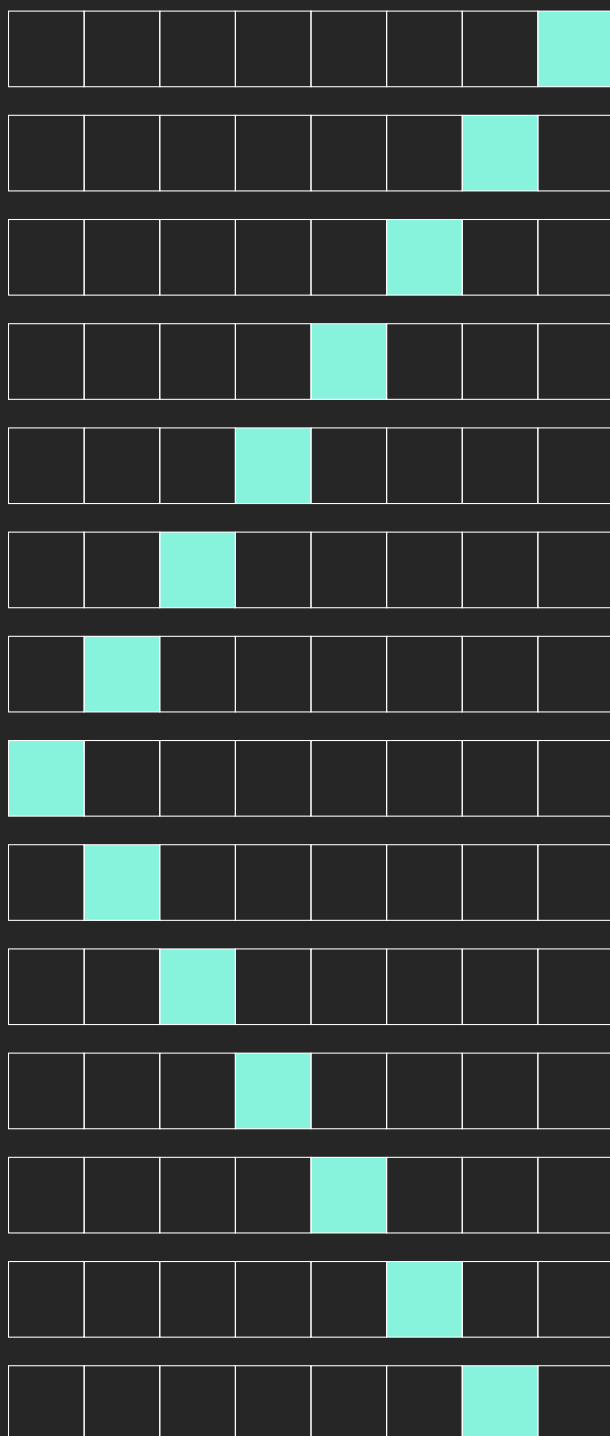
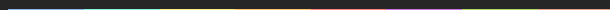


Proyecto 1: Juego de luces

Mario López Sáez

2 de marzo de 2025



Índice

| | | |
|----------|--|----------|
| 1 | Introducción | 3 |
| 2 | Contador parametrizable | 3 |
| 2.1 | Parámetros | 3 |
| 2.2 | Entradas | 3 |
| 2.3 | Salidas | 3 |
| 2.4 | Notas sobre el código | 3 |
| 3 | Pruebas: Contador parametrizable | 4 |
| 3.1 | Parámetros | 4 |
| 3.2 | Funcionamiento normal | 4 |
| 3.3 | Caso extremo 1: Con RST_n y $UP = 0$, ¿se queda TC pillado? | 5 |
| 3.4 | Caso extremo 2: Apagamos ENABLE cuando $TC = 1$ | 5 |
| 3.5 | Vista general | 6 |
| 4 | Registro de desplazamiento parametrizable | 7 |
| 4.1 | Parámetros | 7 |
| 4.2 | Entradas | 7 |
| 4.3 | Salidas | 7 |
| 4.4 | Notas sobre el código | 7 |
| 5 | Pruebas: Registro parametrizable | 7 |
| 5.1 | Parámetros | 7 |
| 5.2 | Simulación RTL | 7 |
| 5.3 | Simulación Gate-Level | 7 |
| 6 | Juego de luces | 8 |
| 6.1 | Parámetros | 8 |
| 6.2 | Entradas | 8 |
| 6.3 | Salidas | 8 |
| 6.4 | Instancias | 8 |
| 6.4.1 | Contador | 8 |
| 6.4.2 | Registro | 8 |
| 6.5 | Notas sobre el código | 8 |
| 7 | Pruebas: Juego de luces | 9 |

3.3 Caso extremo 1: Con RSTn y $UP = 0$, ¿se queda TC pillado?

```
// RSTn con UP = 0 genera un TC prolongado?
UP = 0;
RSTn = 0;
#(T*2)
RSTn = 1;
#(T*4)
UP = 1;
#(T*2)
```

Simulación

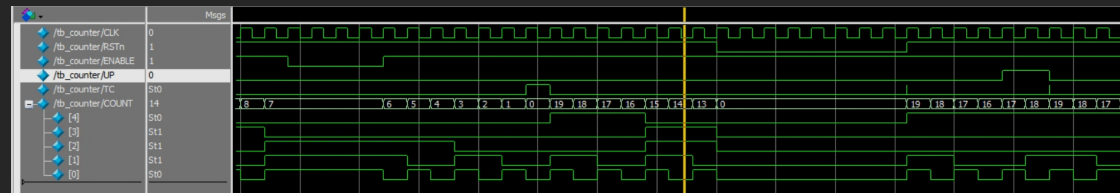


Figura 3: Simulación RTL. Caso extremo 1.

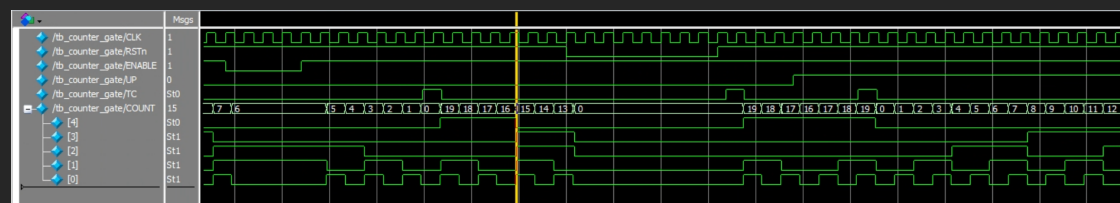


Figura 4: Simulación Gate-Level. TC se activa un ciclo mínimo frente a la delta de RTL.

3.4 Caso extremo 2: Apagamos ENABLE cuando $TC = 1$

```
// ENABLE = 0 cuando TC = 1 deja TC pillado en 1?
wait (duv.TC == 1);
ENABLE = 0;
#(T*4)
ENABLE = 1;
// Lo mismo pero con UP = 0
UP = 0;
#(T*8)
wait (duv.TC == 1);
ENABLE = 0;
#(T*4)
$display("Test finished");
$stop;
end
```

Simulación

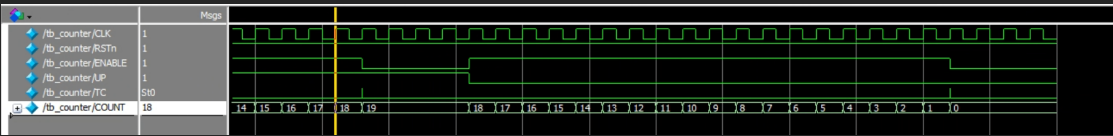


Figura 5: Simulación RTL. Caso extremo 2

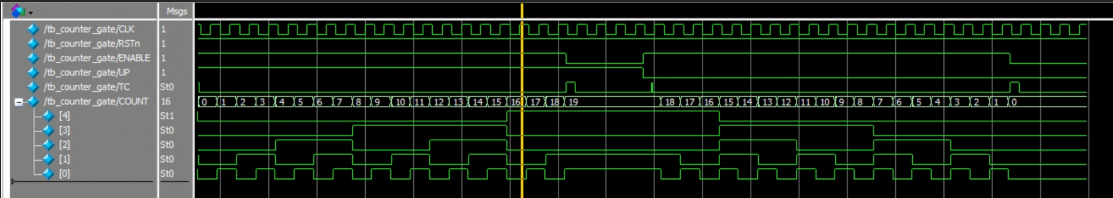


Figura 6: Simulación Gate-Level.

3.5 Vista general

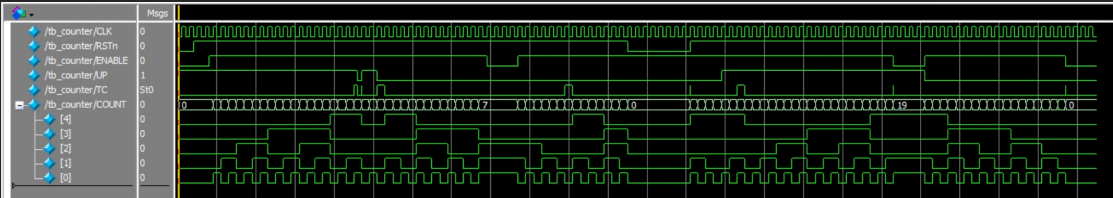


Figura 7: Simulación RTL. Vista general

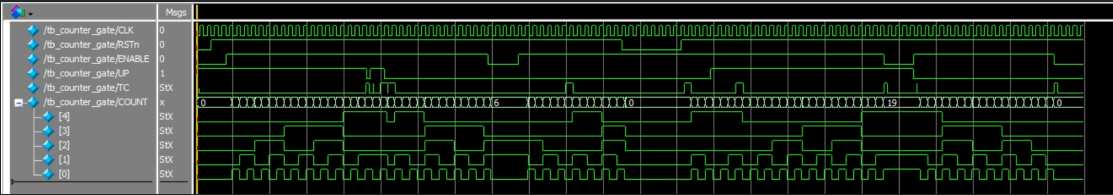


Figura 8: Simulación Gate-Level. Vista general

4 Registro de desplazamiento parametrizable

4.1 Parámetros




Parámetro: ancho

El tamaño de bits del registro.

4.2 Entradas

- Ⓞ Ⓜ CLK — Señal de reloj a 50 MHz
- Ⓞ Ⓜ ENABLE — Entrada síncrona a nivel alto
- Ⓞ Ⓜ SHIFT-IN — Entrada de serie
- Ⓞ Ⓜ RSTn — Reset asíncrono a nivel bajo

4.3 Salidas

- ⌚ →  OUT-P [ancho-1:0] — Salida en paralelo
 →  SHIFT-OUT — Salida de serie

4.4 Notas sobre el código

Always at $\uparrow \odot$ or $\downarrow \text{RSTn}$

Concatenación de señales. Ponemos a la izquierda la entrada SHIFT-IN y a la derecha OUT-P sin incluir OUT-P[0] para mantener el mismo tamaño.

```
OUT_P[ancho-1:0] <= {SHIFT_IN, OUT_P[ancho-1:1]};
```

Assign

Asignamos OUT-P[0] a SHIFT-OUT

5 Pruebas: Registro parametrizable

5.1 Parámetros

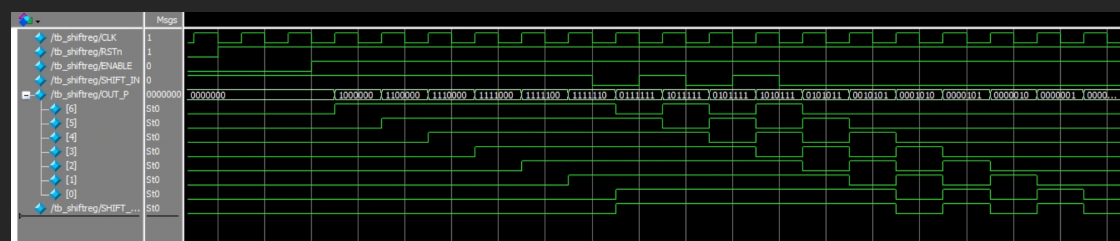
Escala de tiempo

1 ns - Unidad por defecto 100 ps - Precisión de la simulación

Parámetro local: $T = 20 \text{ ns}$

Ancho: 7

5.2 Simulación RTL



5.3 Simulación Gate-Level

