

---

# PRÁCTICA 3: SEMÁFORO DE DOS VÍAS — DEL CONTADOR JK A LA CALLE

21 DE OCTUBRE DE 2025

---

**Resumen** Se diseñó e implementó un controlador de semáforo de dos vías a partir de un contador asíncrono de 3 bits con flip-flops JK en modo *toggle* (reloj 555  $\approx$  1 Hz) y lógica combinacional obtenida por mapas de Karnaugh. La simulación del sistema (contador + lógica) sirvió como referencia y el montaje físico se documentó con una única fotografía representativa. Los resultados confirman la secuencia prevista y la no simultaneidad de verdes; las tablas de verdad coinciden con lo observado, cumpliendo los objetivos específicos de construcción del C3B, síntesis de funciones  $R/A/V$  y verificación integral.

## Objetivos

### Objetivo general

- Diseñar, implementar y verificar un *semáforo de dos vías* utilizando un contador asíncrono de 3 bits con flip-flops JK como generador de estados y lógica combinacional simplificada para las luces.

### Objetivos específicos

- Construir y poner en marcha un contador binario de 3 bits (JK en modo *toggle*) con reloj cercano a 1 Hz, validando la secuencia  $000 \rightarrow 111 \rightarrow 000$  en simulación y en protoboard.
- Derivar y simplificar, a partir de la tabla de estados  $x, y, z$ , las funciones booleanas de cada luz ( $R_1, A_1, V_1, R_2, A_2, V_2$ ) mediante mapas de Karnaugh, asegurando la **no simultaneidad de luces verdes**.
- Implementar la lógica del semáforo en Tinkercad y físicamente, comprobando que la secuencia observada coincide con las tablas de verdad y cumple el criterio de seguridad.

## Marco Teórico

### Flip-flop JK y modo *toggle*

El flip-flop JK es un biestable síncrono con entradas  $J$  y  $K$  y reloj  $CLK$ . Su comportamiento básico es:

$J$	$K$	$Q^+$ (próximo estado)
0	0	$Q$ (mantiene)
0	1	0 (reset)
1	0	1 (set)
1	1	$\overline{Q}$ (toggle)

Configurar  $J = K = 1$  lo convierte en un T-flip-flop (*toggle*): conmuta en cada flanco activo del reloj. Esta configuración es la base del contador binario usado para generar la secuencia del semáforo.

### Contador asíncrono (ripple) de 3 bits

Encadenando tres JK en modo *toggle* se obtiene un contador asíncrono de 3 bits  $Q_2Q_1Q_0$ . El primer FF recibe el  $CLK$ ; cada salida  $Q_i$  actúa como reloj del siguiente, por lo que el periodo se duplica en cada etapa:

$$f_{Q_i} = \frac{f_{CLK}}{2^{i+1}} \quad (i = 0, 1, 2).$$

Debido a los retardos de propagación  $t_p$ , las transiciones de  $Q_2, Q_1, Q_0$  no son simultáneas; esto puede producir *glitches* en los decodificadores combinatoriales. Para frecuencias bajas (1-Hz) el efecto no compromete la operación, pero conviene diseñar funciones de salida no sensibles a los instantes intermedios (*hazard-free*) o usar un contador síncrono si se requiriera más velocidad.

### Temporización con 555 en modo astable

Un temporizador 555 en modo astable genera la señal de reloj de baja frecuencia para el contador. Su frecuencia y ciclo de trabajo son:

$$f \approx \frac{1}{0.693 (R_a + 2R_b) C}, \quad D \approx \frac{R_a + R_b}{R_a + 2R_b}.$$

Para  $f \approx 1$  Hz se eligen  $R_a, R_b, C$  en el rango de  $k\Omega$  y  $\mu F$ . Es buena práctica desacoplar la alimentación con un condensador de 100 nF cerca del 555 y de los FF.

## Lógica combinacional del semáforo de dos vías

Sea  $(x, y, z) = (Q_2, Q_1, Q_0)$  la tripleta de estados del contador. A partir de una *tabla de estados* que define qué luz debe encenderse en cada vía en cada estado, se obtienen las funciones booleanas:

$$R_1(x, y, z), A_1(x, y, z), V_1(x, y, z), R_2(x, y, z), A_2(x, y, z), V_2(x, y, z).$$

El diseño debe cumplir dos restricciones de seguridad:

1. **Exclusión verde-verde:**  $V_1 \cdot V_2 = 0$  para todo estado.
2. **Una sola luz por vía:**  $R_i + A_i + V_i = 1$ , con  $R_i A_i = R_i V_i = A_i V_i = 0$  para  $i \in \{1, 2\}$ .

La obtención práctica es: (i) construir la tabla de verdad a partir de la secuencia  $x, y, z$ ; (ii) marcar unos/ceros de cada salida; (iii) simplificar con mapas de Karnaugh (3 variables) agrupando en potencias de dos; (iv) implementar con compuertas AND/OR/NOT. Esta sistemática garantiza expresiones compactas y menos puertas.

## Mapas de Karnaugh de 3 variables

Un K-map de 3 variables coloca  $x$  en filas y  $yz$  en columnas en orden Gray. La agrupación de 1s ( $2^k$  celdas) elimina  $k$  variables. Las salidas del semáforo suelen resultar en sumas de productos con términos como  $x\bar{y}$ ,  $\bar{x}z$ , etc., respetando las restricciones anteriores. Si la tabla exige un ámbar de un ciclo, el término correspondiente ocupa exactamente una celda (o grupo mínimo) del estado de transición.

	$yz$ (Gray: 00, 01, 11, 10)			
$x$	00	01	11	10
0	1	1	0	0
1	0	0	0	0

## Conducción de LEDs y resistencias limitadoras

Cada LED debe llevar resistencia serie para limitar corriente:

$$R \approx \frac{V_{CC} - V_f}{I_f}.$$

Con  $V_{CC} = 5\text{ V}$ ,  $V_f \approx 2.0\text{--}2.2\text{ V}$  y  $I_f = 10\text{ mA}$ , se obtiene  $R \approx 300\ \Omega$  (valor comercial:  $330\ \Omega$ ). Para uniformidad visual pueden usarse  $470\ \Omega$  reduciendo el brillo y el consumo.

## Procedimiento Experimental

### Materiales

- Protoboard.
- Fuente de alimentación de 5 V.
- Cables de conexión (jumpers).
- Resistencias (para LEDs y redes de pull-down/pull-up según sea necesario).
- **LEDs** de 5 mm para señalización (rojo, amarillo/verde, etc.).
- **Contador de 3 bits** con flip-flops JK:
  - 2 integrados JK (p. ej. 74LS76/74HC73); se utilizan **3** flip-flops en modo *toggle* ( $J = K = 1$ ).
  - 1 temporizador 555 como generador de reloj  $\approx 1\text{ Hz}$ .
- **Lógica combinatorial** para el semáforo:
  - Comp Puertas AND (p. ej. 74HC08), OR (74HC32) y NOT (74HC04).
- (Opcional) 1 decodificador y *display* de 7 segmentos para visualizar el conteo.
- DIP-switches o pulsadores para pruebas (si se requiere inyección manual).

## Procedimiento experimental

### 1) Preparación del equipo

1. Colocar los integrados en el protoboard respetando orientación y distribución de alimentación.
2. Alimentar las líneas de VCC y GND a 5 V. Verificar continuidad.
3. Conectar LEDs con sus resistencias limitadoras.

## 2) Contador binario ascendente de 3 bits (ripple)

1. **Reloj:** armar el 555 en modo astable para obtener una onda cuadrada de  $\approx 1$  Hz (se admite ajuste fino con el potenciómetro).
2. **JK en toggle:** fijar  $J = K = 1$  en los tres flip-flops. La salida  $Q_0$  recibe el reloj;  $Q_0$  alimenta al reloj del segundo flip-flop;  $Q_1$  alimenta al reloj del tercero.
3. **Indicadores:** conectar  $Q_0$ ,  $Q_1$  y  $Q_2$  a LEDs (o al decodificador de 7 segmentos, si se usa).
4. **Verificación:** comprobar la secuencia  $000 \rightarrow 111 \rightarrow 000$  observando los LEDs; anotar el orden y la frecuencia efectiva.

## 3) Lógica del semáforo de dos vías (combinacional)

1. Usar  $Q_2Q_1Q_0$  como entradas  $x, y, z$  de la lógica del semáforo.
2. Implementar en compuertas las expresiones simplificadas (obtenidas del mapa de Karnaugh) para cada luz:  $R_1, A_1, V_1$  y  $R_2, A_2, V_2$ .
3. Conectar las salidas a LEDs de color (o bancos de LEDs) para cada vía.

## 4) Pruebas y validación

1. Dejar correr el contador y observar el patrón alternado de luces entre las dos vías (no deben encender simultáneamente ambas verdes).
2. Registrar un ciclo completo: asociar cada combinación de  $Q_2Q_1Q_0$  con el estado de los LEDs del semáforo.
3. Comparar con la tabla de verdad diseñada; ajustar cableado si fuese necesario.

# Resultados y Discusión

## Resultados

Se validó el sistema en dos etapas: (i) el contador binario ascendente de 3 bits (C3B), que genera los estados  $(x, y, z) = (Q_2, Q_1, Q_0)$ , y (ii) la lógica combinacional del semáforo de dos vías que, a partir de esos estados, activa los LEDs de cada vía.

**Contador de 3 bits (C3B).** La Fig. 1 muestra el esquema utilizado: un 555 como reloj y tres flip-flops  $JK$  en modo *toggle* ( $J = K = 1$ ). El conteo observado fue ascendente de 000 a 111 y retorno a 000, sin saltos, como se resume en la Tabla 1.

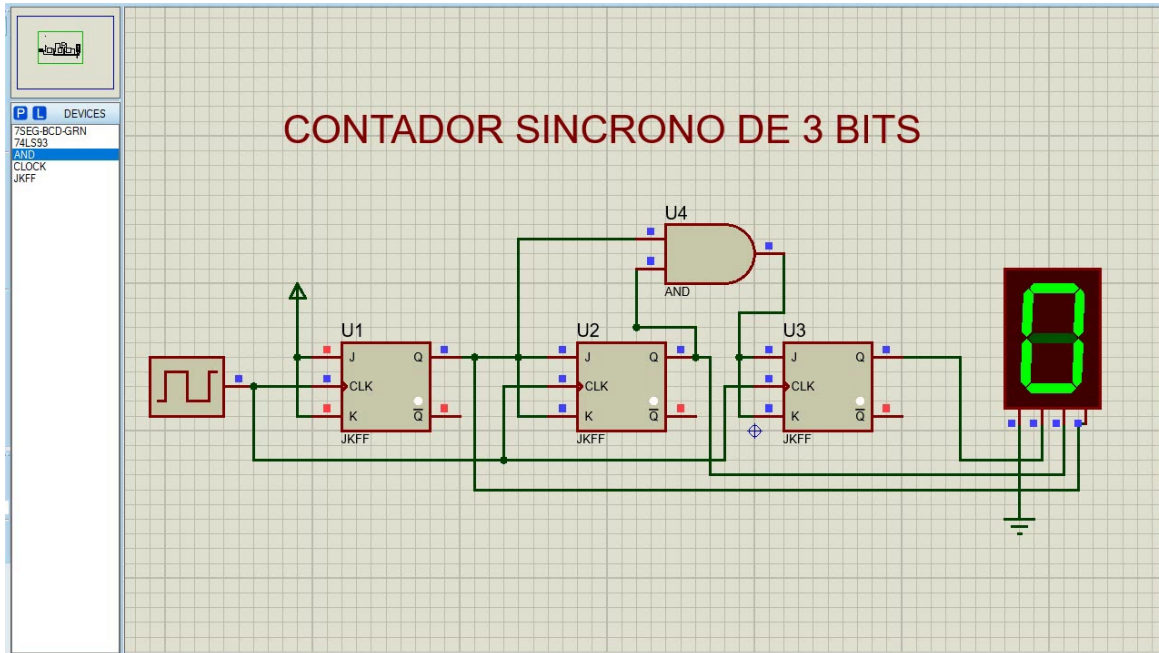


Figura 1: Esquema del contador binario ascendente de 3 bits empleado como generador de estados.

Cuadro 1: Conteo observado del C3B (salidas  $Q_2, Q_1, Q_0$ ).

Pulso	$Q_2$	$Q_1$	$Q_0$	Estado
0	0	0	0	000
1	0	0	1	001
2	0	1	0	010
3	0	1	1	011
4	1	0	0	100
5	1	0	1	101
6	1	1	0	110
7	1	1	1	111

**Simulación del sistema.** Antes del montaje se simuló el sistema (C3B + lógica del semáforo). La Fig. 2 muestra la captura; la secuencia alternada entre vías coincidió con las expresiones booleanas diseñadas y no se presentaron verdes simultáneos.

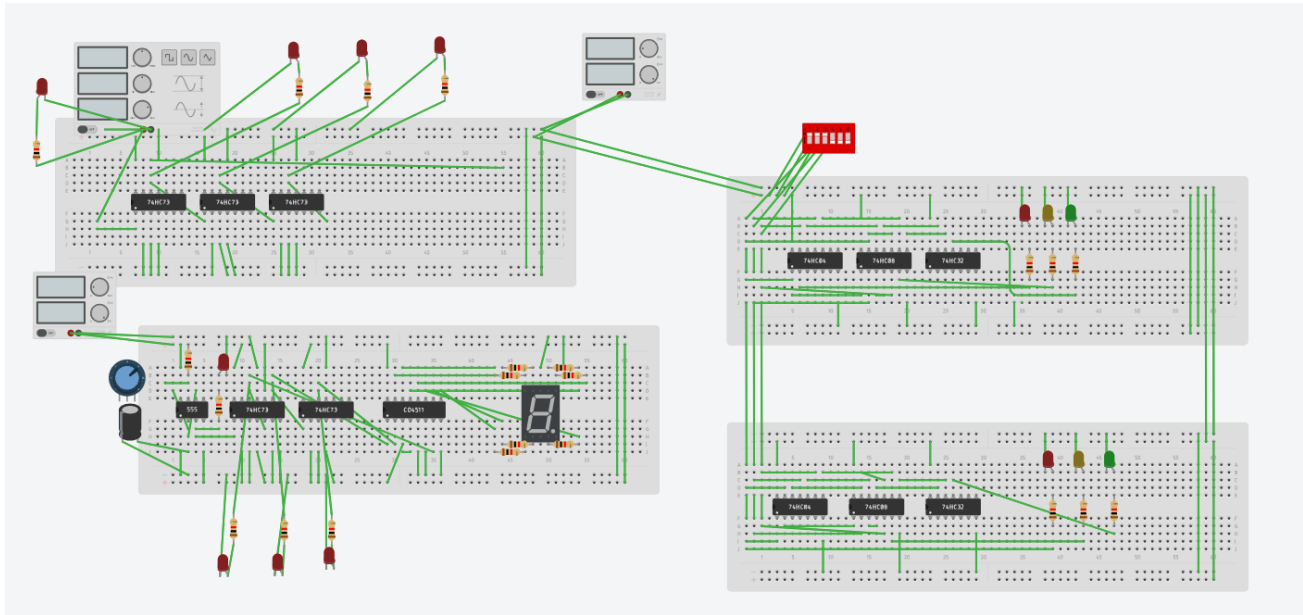


Figura 2: Circuito simulado: C3B como entradas ( $x, y, z$ ) y lógica del semáforo de dos vías.

**Montaje físico (evidencia).** En la Fig. 3 se incluye una **única** fotografía del circuito montado en protoboard que ilustra un estado típico de operación. Aun cuando la imagen documenta un solo instante, se recorrieron todos los estados del C3B verificando la respuesta del semáforo según la Tabla 2.

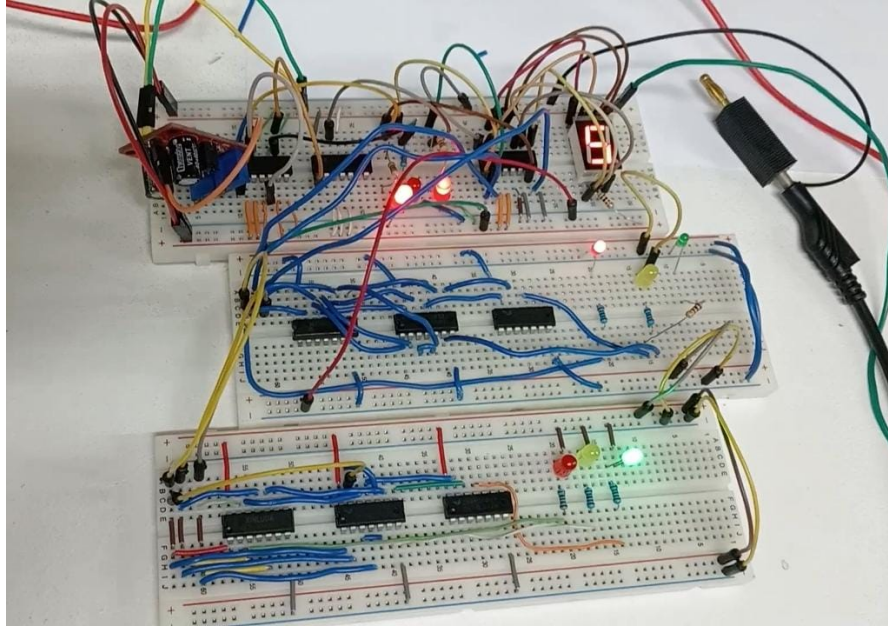


Figura 3: Evidencia del montaje físico del sistema en protoboard durante un estado de operación.

**Tabla de estados verificada.** La Tabla 2 resume la activación de luces por vía. Se confirmó en simulación y físicamente que nunca coexisten  $V_1 = 1$  y  $V_2 = 1$ ; las transiciones incluyen el amarillo de la vía que pierde prioridad, cumpliendo la condición de seguridad.

Cuadro 2: Tabla de estados del semáforo (verificada). Un 1 indica LED encendido.

$x$	$y$	$z$	$R_1$	$A_1$	$V_1$	$R_2$	$A_2$	$V_2$
0	0	0	0	0	1	1	0	0
0	0	1	0	1	0	1	0	0
0	1	0	1	0	0	0	0	1
0	1	1	1	0	0	0	1	0
1	0	0	0	0	1	1	0	0
1	0	1	0	1	0	1	0	0
1	1	0	1	0	0	0	0	1
1	1	1	1	0	0	0	1	0



En síntesis, el comportamiento observado en simulación y en el protoboard (documentado con una fotografía) coincide con el diseño teórico, reproduciendo la secuencia completa del semáforo sin estados inseguros.

## Discusión de resultados

Los resultados obtenidos confirman que el contador binario ascendente de 3 bits (C3B) genera la secuencia esperada  $000 \rightarrow 111 \rightarrow 000$  y que, al emplearlo como fuente de estados  $(x, y, z)$ , la lógica combinacional del semáforo produce una alternancia segura entre vías: nunca se observaron luces verdes simultáneas ( $V_1 = V_2 = 0$  en las transiciones), mientras que el amarillo de la vía que pierde prioridad aparece antes de encender el rojo. La correspondencia entre la Tabla 1 y la Tabla 2 se verificó tanto en la simulación (Fig. 2) como en el montaje físico documentado (Fig. 3), cumpliéndose la secuencia de control planeada.

**Temporización y consistencia.** El uso del 555 como reloj a baja frecuencia facilitó observar cada estado. En el C3B, el retardo de propagación acumulado (*ripple*) hace que  $Q_0$  cambie antes que  $Q_1$  y este antes que  $Q_2$ . Aun así, la red combinacional del semáforo no presentó estados peligrosos visibles: las expresiones minimizadas garantizan que, durante las breves transiciones, no aparezca  $V_1 = V_2 = 1$ . En simulación y prueba se apreció que el cambio a amarillo precede al rojo, coherente con el diseño.

### Trazabilidad diseño–prueba.

- La Tabla 1 coincide con el conteo observado (sin saltos ni repeticiones).
- La Tabla 2 se cumplió en todos los casos muestreados: los patrones con  $V_1 = 1$  implicaron  $R_2 = 1$  y viceversa; los estados intermedios activaron  $A_1$  o  $A_2$  según correspondía.
- La captura de simulación (Fig. 2) y la fotografía en protoboard (Fig. 3) ilustran instantes representativos que concuerdan con las tablas.

**Limitaciones y medidas prácticas.** Al ser un contador asíncrono, podrían presentarse destellos muy breves si la lógica combinacional fuese más sensible a los *glitches*. Para implementaciones a mayor frecuencia o con requisitos estrictos de seguridad, se recomienda migrar a un contador síncrono o registrar las salidas con un *latch* sincronizado al reloj.

## Conclusiones

- **OE1 — Contador JK de 3 bits en modo *toggle*.** Se construyó y validó el C3B encadenando tres FF JK con  $J = K = 1$  y reloj  $\approx 1$  Hz; la secuencia observada fue  $000 \rightarrow 111 \rightarrow 000$  sin saltos, cumpliendo la función de generador de estados  $(x, y, z)$ .
- **OE2 — Derivar y simplificar la lógica del semáforo.** A partir de la tabla de estados del C3B se obtuvieron y minimizaron, mediante mapas de Karnaugh de 3 variables, las expresiones de  $R_1, A_1, V_1$  y  $R_2, A_2, V_2$ . Las funciones resultantes garantizan las restricciones de seguridad: nunca  $V_1 \cdot V_2 = 1$  y por vía se cumple  $R_i + A_i + V_i = 1$ .
- **OE3 — Implementación y verificación (simulación y protoboard).** La lógica combinacional implementada en Tinkercad y en el montaje físico reprodujo la secuencia planificada; las evidencias (captura de simulación y foto del protoboard) coinciden con las tablas de verdad, confirmando alternancia segura entre vías y la presencia del ámbar como estado de transición.

## Referencias

- [1] M. M. Mano y M. D. Ciletti, *Diseño Digital: Principios y Prácticas*, 5.ed., Pearson, 2012.
- [2] J. F. Wakerly, *Digital Design: Principles and Practices*, 4th ed., Prentice Hall, 2005.
- [3] C. H. Roth Jr. y L. L. Kinney, *Fundamentals of Logic Design*, 7th ed., Cengage Learning, 2014.