

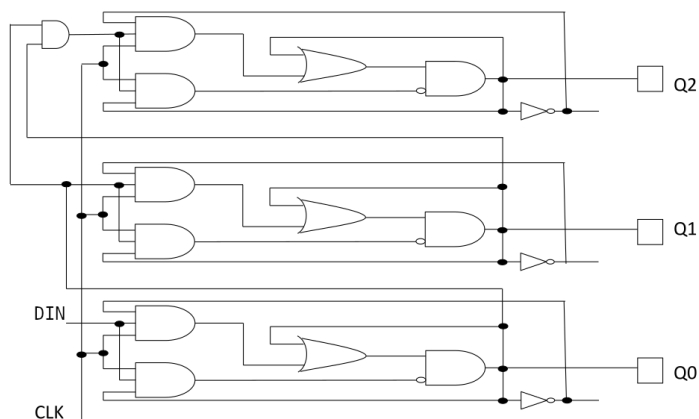
# Sobre la derivación de un modelo matemático representativo del registro contador de 3 bits

Savinelli Roberto Nicolás  
[rsavinelli@est.frba.utn.edu.ar](mailto:rsavinelli@est.frba.utn.edu.ar)

4 de Julio de 2020

## 1. Introducción

Partiendo del circuito esquemático de un registro contador de 3 bits representado en la figura 1, se procede a deducir el modelo matemático que explica el comportamiento del mismo, con el fin de poder calcular los valores de  $Q_0$ ,  $Q_1$  y  $Q_2$  para  $CLK = 0$  y  $CLK = 1$  cuando  $DIN = 0$  y  $DIN = 1$ .



**Figura 1:** Registro contador de 3 bits

Para lograr tal fin, fue necesario el conocimiento de de herramientas tales como el álgebra de Boole y la lógica digital, los circuitos secuenciales, los sistemas de representación numéricos, y el marco teórico que la representación esquemática de circuitos implica.

Con el propósito de que este documento no sea de una extensión irrisoria para su cometido, se omitirán las definiciones de los conceptos anteriormente mencionados, y se hará la asunción de que el lector ha visto, estudiado, o está familiarizado con los mismos.

Sin más preámbulo, se procede al desarrollo del modelo.

## 2. Caso $CLK = 0$

Denomínense, los valores anteriores de los registros, como  $Q_0 anterior$ ,  $Q_1 anterior$ , y  $Q_2 anterior$ . Y, los valores nuevos de los registros, como  $Q_0$ ,  $Q_1$  y  $Q_2$ .

Independientemente del valor de  $DIN$ , para  $CLK = 0$ , se mantienen los valores de  $Q_0 anterior$ ,  $Q_1 anterior$ , y  $Q_2 anterior$ , previamente almacenados en los registros.

Esto es, si previo a pasar de  $CLK = 1$  a  $CLK = 0$ , los valores de los registros fueran:

$$Q_0 anterior = a \quad Q_1 anterior = b \quad Q_2 anterior = c \quad (1)$$

Entonces, los valores de  $Q_0$ ,  $Q_1$ , y  $Q_2$ , sería  $a$ ,  $b$ , y  $c$  respectivamente.

$$Q_0 = a \quad Q_1 = b \quad Q_2 = c \quad (2)$$

¿Cómo se llegó a esta conclusión?

Tomando como segmento ejemplar de la lógica detrás del circuito al biestable inferior que  $Q_0$  comprende, y nombrando a cada componente del mismo de la siguiente manera:

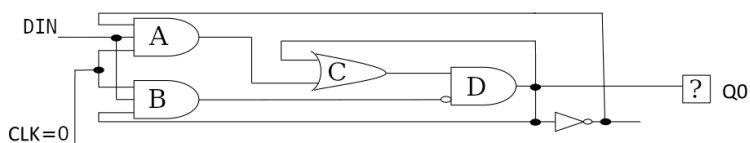


Figura 2: Flip-Flop JK síncronico

Se procede a explicar la lógica detrás de este razonamiento.

### 2.1. Compuertas Lógicas A y B

Con el fin de simplificar el circuito, analicemos primero, de forma genérica, cómo se comporta una compuerta lógica AND de tres entradas apoyándonos en el esquema de la compuerta B, que toma como valores de entrada a  $CLK$ ,  $DIN$  y  $Q_0 anterior = a$ :



Figura 3: Compuerta Lógica AND B

Sabemos que, la función producto lógico, o función AND, es verdadera sólo cuando todas las variables de entrada son iguales a 1. Siendo que, en el circuito, las compuertas que toman como valor de entrada a  $CLK$  son de este tipo (con tres valores de entrada), vemos que, el único caso en que  $CLK \cdot DIN \cdot a$  es distinto de cero es cuando  $CLK = DIN = a = 1$  (véase tabla I en el apéndice B para más detalle).

Podemos afirmar, entonces, que con  $CLK = 0$  los valores de salida tanto de  $A$  como de  $B$  van a ser iguales a cero sin importar qué valores tomen las otras entradas.

$$A = 0 \quad B = 0 \quad (3)$$

## 2.2. Compuerta Lógica C

Obtenidos  $A$  y  $B$  podemos proceder a calcular el valor de la compuerta lógica  $C$ .

Partiendo de la misma premisa que el caso anterior analicemos, primero, cómo se comporta una compuerta lógica OR de dos entradas utilizando la nomenclatura del circuito para con  $C$ , siendo los valores de entrada, de la misma, el valor de salida de  $A$  y  $Q_0 anterior = a$ :



**Figura 4:** Compuerta Lógica OR C

Sabemos que, la función suma lógica, o función OR, es verdadera cuando alguna de las variables de entrada es verdadera, o en otras palabras, cuando su valor de verdad es igual a 1.

Entonces, siendo los valores de interés para esta operación

$$A = 0 \quad Q_0 anterior = a \quad (4)$$

por ser, los mismos, los valores que la compuerta  $C$  toma como valores de entrada. Entonces:

$$\begin{aligned} C &= A \ || \ Q_0 anterior \\ C &= 0 \ || \ a \\ C &= a \end{aligned} \quad (5)$$

## 2.3. Compuerta Lógica D

La compuerta  $D$  toma como valores de entrada el valor de salida de  $C$ , y el valor negado de la salida de  $B$ , ergo, los valores de carácter necesario para llevar a cabo esta operación son:

$$C = a \quad B = 0 \quad (6)$$

Planteando, entonces, la propoción lógica de  $D$ :

$$\begin{aligned} D &= C \ \&\& \ \sim B \\ D &= a \ \&\& \ \sim 0 \\ D &= a \ \&\& \ 1 \end{aligned}$$

Observamos en este último paso que, el valor de  $D$ , dependerá únicamente de qué valor tome  $a$ . Separando los dos casos posibles tenemos i) para  $a = 0$  entonces,  $D = 0$  y, ii) para  $a = 1$  entonces,  $D = 1$ .

Esto es,

$$D = a \quad (7)$$

Como  $Q_0 \text{ anterior} = a$ ,

$$Q_0 = D = Q_0 \text{ anterior} = a \quad (8)$$

Comprobando así que, independientemente del valor de  $DIN$ , para  $CLK = 0$ , se mantienen el valor de  $Q_0 \text{ anterior}$ , previamente almacenado en el registro.

Ahora bien, extendiendo esta demostración a los dos circuitos Flip-Flop JK sincrónicos restantes que componen al registro contador de tres bits en estudio, se puede demostrar que los valores nuevos de los registros se mantienen iguales a los previamente almacenados en los mismos.

### 3. Caso CLK = 1

¿Qué pasaría si, en este escenario, emitiésemos la señal  $CLK = 1$  con  $DIN = 0$ ?

Para esta interrogante, se propone al lector emular el procedimiento utilizado para el caso en que  $CLK = 0$  pero, en esta ocasión, utilizando  $DIN = 0$ .

¿Y si DIN fuera igual a 1?

Para la resolución de este caso, y para la confección del cuadro 1 se decidió continuar con el enfoque utilizado para el caso  $CLK = 0$ , y tomar los valores desconocidos de  $Q_0$  anterior,  $Q_1$  anterior y  $Q_2$  anterior como  $a$ ,  $b$  y  $c$ , y en base a ello se calcularon los valores para  $DIN = 1$ .

Tomando a los valores nuevos de los registros como  $Q_0$ ,  $Q_1$ , y  $Q_2$ , entonces, los valores de las incógnitas de cada biestable, en forma genérica, se calculan según las operaciones lógicas presentes en el siguiente cuadro:

Registro	Operación lógica para la obtención de su valor
$Q_0$	$(\sim Q_0 \text{ anterior})$
$Q_1$	$[(\sim Q_0) \vee (Q_1 \text{ anterior})] \wedge [(Q_0) \vee (\sim Q_1 \text{ anterior})]$
$Q_2$	$[(\sim Q_0) \vee (Q_2 \text{ anterior})] \wedge [(\sim Q_1) \vee (Q_2 \text{ anterior})] \wedge [(Q_0) \vee (Q_1) \vee (\sim Q_2 \text{ anterior})]$

**Cuadro 1:** Operaciones lógicas para obtener  $Q_0$ ,  $Q_1$  y  $Q_2$

Nótese como, al igual que lo realiza el circuito, se utiliza el valor actualizado de las variables  $Q_0$ ,  $Q_1$ , y  $Q_2$  para realizar las operaciones, y no siempre los iniciales.

Un ejemplo de cómo aplicar las fórmulas presentes en el cuadro 1, y del paso a paso para la obtención del valor de los registros pueden verse en el Apéndice A al final del presente documento.

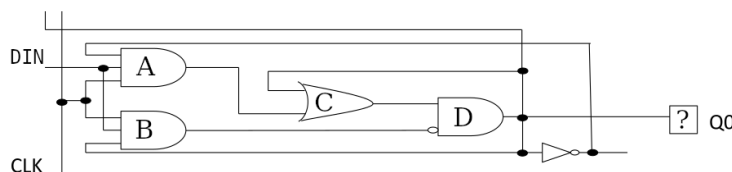
Para explicar y dar fundamento a los resultados obtenidos, se procederá a calcular, de manera segmentada, el resultado de salida de cada uno de los Flip-Flop JK sincrónicos.

#### 3.1. Primer segmento

Tomando como segmento inicial el inferior, denominando los resultados de salida de cada componente del circuito esquemático como  $A$ ,  $B$ ,  $C$  y  $D$  (véase figura 5), y siendo los valores de carácter necesarios para este primer segmento:

$$DIN = 1 \quad CLK = 1 \quad Q_0 \text{ anterior} = a \tag{1}$$

Se detalla, entonces, el proceso aritmético llevado a cabo.



**Figura 5:** Flip-Flop JK sincrónico primero

### 3.1.1. Compuerta lógica A

Siendo la misma una compuerta lógica AND de 3 entradas,  $DIN$ ,  $CLK$ , y  $\sim Q_0 anterior$ , obtenemos que el valor de salida de  $A$  va ser igual a:

$$\begin{aligned}A &= (DIN) \&\& (CLK) \&\& (\sim Q_0 anterior) \\A &= (1) \&\& (1) \&\& (\sim a) \\A &= (\sim a)\end{aligned}\tag{2}$$

### 3.1.2. Compuerta lógica B

Mismo caso para la compuerta B. Esta es una compuerta lógica AND de 3 entradas,  $DIN$ ,  $CLK$ , y  $Q_0 anterior$ . Representando los datos en forma analítica, obtenemos que el valor de salida de  $B$  será:

$$\begin{aligned}B &= (DIN) \&\& (CLK) \&\& (Q_0 anterior) \\B &= (1) \&\& (1) \&\& (a) \\B &= (a)\end{aligned}\tag{3}$$

### 3.1.3. Compuerta lógica C

Calculado el valor de salida de la compuerta  $A$ , y conocido el valor de  $Q_0 anterior$ , se procede a calcular el valor de salida de la compuerta lógica OR  $C$ , que toma a los mismos como valores de entrada:

$$\begin{aligned}C &= (Q_0 anterior) \|\| (A) \\C &= (a) \|\| (\sim a) \\C &= (1)\end{aligned}\tag{4}$$

### 3.1.4. Compuerta lógica D

Finalmente, obtenidos los valores de  $B$  y  $C$ , podemos calcular el valor de salida de la compuerta lógica AND que representa  $D$ .

Esto es,

$$\begin{aligned}D &= (C) \&\& (\sim B) \\D &= (1) \&\& (\sim a) \\D &= (\sim a)\end{aligned}\tag{5}$$

Quedando como resultado de esta operatoria:

$$Q_0 = D = (\sim Q_0 anterior)\tag{6}$$

### 3.2. Segundo segmento

Para este segundo segmento se denominaron, los resultados de salida de cada componente del circuito esquemático,  $E$ ,  $F$ ,  $G$  y  $H$  (véase figura 6), y siendo los valores de carácter necesarios para este segundo segmento:

$$CLK = 1 \quad Q_0 \text{ anterior} = a \quad Q_1 \text{ anterior} = b \quad (7)$$

Se detalla, entonces, el proceso aritmético llevado a cabo.

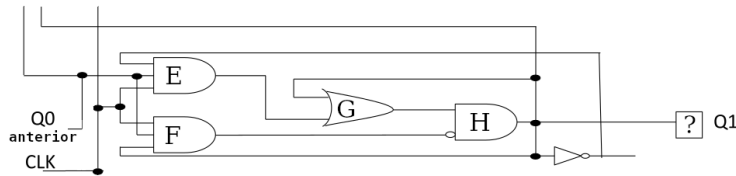


Figura 6: Flip-Flop JK síncrono segundo

#### 3.2.1. Compuerta lógica E

Siendo la misma una compuerta lógica AND de 3 entradas,  $\sim Q_1$  anterior,  $CLK$ , y  $Q_0$  anterior, obtenemos que el valor de salida de  $E$  va ser igual a:

$$\begin{aligned} E &= (\sim Q_1 \text{ anterior}) \&\& (CLK) \&\& (Q_0 \text{ anterior}) \\ E &= (\sim b) \&\& (1) \&\& (a) \\ E &= (\sim b) \&\& (a) \end{aligned} \quad (8)$$

#### 3.2.2. Compuerta lógica F

Compuerta lógica AND de 3 entradas,  $CLK$ ,  $Q_0$  anterior, y  $Q_1$  anterior, esto es:

$$\begin{aligned} F &= (CLK) \&\& (Q_0 \text{ anterior}) \&\& (Q_1 \text{ anterior}) \\ F &= (1) \&\& (a) \&\& (b) \\ F &= (a) \&\& (b) \end{aligned} \quad (9)$$

#### 3.2.3. Compuerta lógica G

Habiendo sido  $E$  calculado previamente, se procede a calcular el valor de  $G$  para la compuerta lógica OR de 2 entradas,  $Q_1$  anterior y  $E$ :

$$\begin{aligned} G &= (E) \parallel (Q_1 \text{ anterior}) \\ G &= [(\sim b) \&\& (a)] \parallel b \\ G &= [(\sim b) \parallel b] \&\& [(a) \parallel b] \\ G &= (1) \&\& [(a) \parallel b] \\ G &= (a) \parallel b \end{aligned} \quad (10)$$

### 3.2.4. Compuerta lógica H

Finalmente, obtenidos los valores de  $F$  y  $G$ , podemos proceder a calcular el valor de salida de la compuerta lógica AND que  $H$  representa:

$$\begin{aligned}
 H &= (G) \&\& (\sim F) \\
 H &= [(a) \parallel b] \&\& \sim [(a) \&\& (b)] \\
 H &= [(a) \parallel b] \&\& [(\sim a) \parallel (\sim b)]
 \end{aligned}
 \tag{11}$$

Ahora bien, volviendo a la premisa de que:

$$Q_0 \text{ anterior} = a$$

Y que, como se ha calculado anteriormente:

$$Q_0 = \sim Q_0 \text{ anterior:}$$

Entonces:

$$\begin{aligned}
 \sim Q_0 &= \sim \sim Q_0 \text{ anterior} \\
 \sim Q_0 &= Q_0 \text{ anterior} \\
 \sim Q_0 &= a
 \end{aligned}$$

Además, siendo  $b$  representativo de  $Q_1 \text{ anterior}$ , entonces, queda como resultado de esta operatoria:

$$Q_1 = H = [(\sim Q_0) \parallel (Q_1 \text{ anterior})] \&\& [(Q_0) \parallel (\sim Q_1 \text{ anterior})] \tag{12}$$

Este cambio en la notación nos permite utilizar el valor, previamente calculado como valor de salida  $Q_0$ , para la obtención del valor nuevo del registro  $Q_1$ .

### 3.3. Tercer Segmento

Tomando como segmento final el superior, denominando los resultados de salida de cada componente del circuito esquemático como  $I, J, K, L$  y  $M$  (véase figura 7), y siendo los valores de carácter necesarios para este primer segmento:

$$CLK = 1 \quad Q_2 \text{ anterior} = c \quad Q_1 \text{ anterior} = b \quad Q_0 \text{ anterior} = a \tag{13}$$

Se detalla, entonces, el proceso aritmético llevado a cabo.

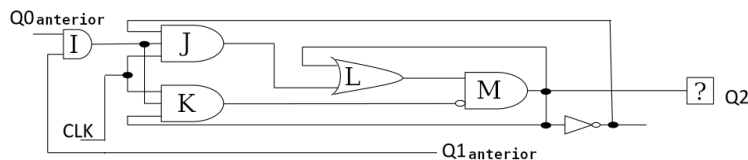


Figura 7: Flip-Flop JK síncrono tercero



### 3.3.1. Compuerta Lógica I

Tomando como valores de entrada los valores de  $Q_0$  anterior y  $Q_1$  anterior, el valor de salida de la compuerta lógica que  $I$  representa es:

$$\begin{aligned} I &= (Q_0 \text{ anterior}) \&\& (Q_1 \text{ anterior}) \\ I &= (a) \&\& (b) \end{aligned} \quad (14)$$

### 3.3.2. Compuerta Lógica J

Ahora bien, siendo  $J$  una compuerta lógica AND con parametros de entrada  $\sim Q_2$  anterior,  $I$ , y  $CLK$ , entonces:

$$\begin{aligned} J &= (\sim Q_2 \text{ anterior}) \&\& (I) \&\& (CLK) \\ J &= (\sim c) \&\& (a) \&\& (b) \&\& (1) \\ J &= (\sim c) \&\& (a) \&\& (b) \end{aligned} \quad (15)$$

### 3.3.3. Compuerta Lógica K

Para  $K$ , compuerta lógica AND, con parametros de entrada  $CLK$ ,  $I$ , y  $Q_2$  anterior, obtenemos:

$$\begin{aligned} K &= (Q_2 \text{ anterior}) \&\& (I) \&\& (CLK) \\ K &= (c) \&\& (a) \&\& (b) \&\& (1) \\ K &= (c) \&\& (a) \&\& (b) \end{aligned} \quad (16)$$

### 3.3.4. Compuerta Lógica L

Resuelta la compuerta  $J$  y conociendo  $Q_2$  anterior, para la compuerta lógica OR  $L$ , que toma los parámetros ya mencionados, entonces:

$$\begin{aligned} L &= (J) \parallel (Q_2 \text{ anterior}) \\ L &= [(\sim c) \&\& (a) \&\& (b)] \parallel (c) \\ L &= [(\sim c) \parallel (c)] \&\& [(a) \parallel (c)] \&\& [(b) \parallel (c)] \\ L &= (1) \&\& [(a) \parallel (c)] \&\& [(b) \parallel (c)] \\ L &= [(a) \parallel (c)] \&\& [(b) \parallel (c)] \\ L &= [(a) \&\& (b)] \parallel (c) \end{aligned} \quad (17)$$

### 3.3.5. Compuerta Lógica M

Tomando como parámetros los valores de salida de  $L$  y  $\sim K$  como entrada, el valor de salida de la compuerta lógica AND  $M$  deviene:

$$\begin{aligned} M &= (L) \&\& (\sim K) \\ M &= [[(a) \&\& (b)] \parallel (c)] \&\& \sim [(c) \&\& (a) \&\& (b)] \\ M &= [[(a) \parallel (c)] \&\& [(b) \parallel (c)]] \&\& [(\sim c) \parallel (\sim a) \parallel (\sim b)] \end{aligned}$$

$$M = [(a \parallel (c)) \&\& [(b \parallel (c)) \&\& [(\sim a) \parallel (\sim b) \parallel (\sim c)]] \quad (18)$$

Resultando finalmente de aplicar un procedimiento análogo al utilizado para calcular  $a$  en función de  $Q_0$  en el segundo segmento, y sabiendo que  $c = Q_2$  anterior:

$$Q_2 = M = [(\sim Q_0) \parallel (Q_2 \text{ anterior})] \&\& [(\sim Q_1) \parallel (Q_2 \text{ anterior})] \&\& [(Q_0) \parallel (Q_1) \parallel (\sim Q_2 \text{ anterior})] \quad (19)$$

Nótese que para este caso se optó por no expandir  $Q_1$  en su forma

$$Q_1 = H = [(\sim Q_0) \parallel (Q_1 \text{ anterior})] \&\& [(Q_0) \parallel (\sim Q_1 \text{ anterior})]$$

con el objetivo de no hacer ininteligible la fórmula respectiva a  $Q_2$  ya que, del mismo modo que la fórmula despejada para  $Q_1$  utiliza el valor obtenido de  $Q_0$ , esta notación nos permite utilizar los valores nuevos calculados de los registros que preceden a  $Q_2$  para este mismo fin.

Véase además cómo, para el caso de  $Q_1$ , se emuló el despeje realizado para utilizar  $\sim Q_0$  en remplazo de  $a$  en el segundo segmento. Esto es posible debido a que, cada Flip-Flop JK sincrónico respeta el mismo modelo matemático, y persigue el mismo comportamiento lógico. La diferencia radica en que, al estar conectadas las salidas de unos a las entradas otros, la expresión del valor del registro evaluado, además, estará condicionado por la lógica del precedente.

## Apéndice A

### Ejemplo aplicado

Tomemos el valor decimal 5, cuya representación en binario utilizando sólo 3 bits es 101. En este caso, al introducir el número en los registros obtendríamos que, como punto de partida, tenemos:

$$Q_0 \text{ anterior} = 1 \quad Q_1 \text{ anterior} = 0 \quad Q_2 \text{ anterior} = 1$$

Analicemos el primer registro, tenemos:

$$CLK = 1 \quad DIN = 1 \quad Q_0 \text{ anterior} = 1$$

Por lo tanto, utilizando el cuadro 1 provisto para el caso  $CLK = 1$ , obtenemos que el valor de  $Q_0$  es:

$Q_0$	$(\sim Q_0 \text{ anterior})$
$Q_0$	$(\sim 1)$
$Q_0$	$(0)$

Con el valor obtenido podemos proceder a calcular  $Q_1$ :

$Q_1$	$[(\sim Q_0) \vee (Q_1 \text{ anterior})] \&\& [(Q_0) \vee (\sim Q_1 \text{ anterior})]$
$Q_1$	$[(\sim 0) \vee (0)] \&\& [(0) \vee (\sim 0)]$
$Q_1$	$[(1) \vee (0)] \&\& [(0) \vee (1)]$
$Q_1$	$(1) \&\& (1)$
$Q_1$	$(1)$

Y con este,  $Q_2$ :

$Q_2$	$[(\sim Q_0) \vee (Q_2 \text{ anterior})] \&\& [(\sim Q_1) \vee (Q_2 \text{ anterior})] \&\& [(Q_0) \vee (Q_1) \vee (\sim Q_2 \text{ anterior})]$
$Q_2$	$[(\sim 0) \vee (1)] \&\& [(\sim 1) \vee (1)] \&\& [(0) \vee (1) \vee (\sim 1)]$
$Q_2$	$[(1) \vee (1)] \&\& [(0) \vee (1)] \&\& [(0) \vee (1) \vee (0)]$
$Q_2$	$(1) \&\& (1) \&\& (1)$
$Q_2$	$(1)$

De aquí que los valores de salida sean:

$$Q_0 = 0 \quad Q_1 = 1 \quad Q_2 = 1$$

Finalmente, colocamos cada valor de registro en su posición correspondiente para la representación binaria de 3 bits. Esto es, de derecha a izquierda,  $Q_0$  en la primera posición,  $Q_1$  en la segunda y  $Q_2$  en la tercera:

$$\begin{array}{ccc} Q_2 & Q_1 & Q_0 \\ 1 & 1 & 0 \end{array}$$

Siendo, el número obtenido en binario 110 (número 6 en decimal) que inmediatamente precede al 101.

Con esto, queda ejemplificado y modelizado el funcionamiento del registro contador de 3 bits para el caso en el que los valores iniciales, de dichos registros, sean los dados al principio de esta demostración y, los valores de entrada  $CLK$  y  $DIN$ , sean iguales a 1.

De manera análoga puede describirse el comportamiento del circuito con el resto de los números que el registro 3 bits permite.

## Análisis del límite superior

¿Qué pasaría en el caso en que todos los valores anteriores de los registros fuesen iguales a 1 y se reciban las señales  $CLK = DIN = 1$ ?

Para este caso, los valores iniciales de los registro son:

$$Q_0 \text{ anterior} = 1 \quad Q_1 \text{ anterior} = 1 \quad Q_2 \text{ anterior} = 1$$

Y los valores de interés para calcular  $Q_0$  son:

$$CLK = 1 \quad DIN = 1 \quad Q_0 \text{ anterior} = 1$$

Aplicando el mismo método utilizado en el ejemplo anterior:

$Q_0$	$(\sim Q_0 \text{ anterior})$
$Q_0$	$(\sim 1)$
$Q_0$	$(0)$

Obtenido  $Q_0$ , se procede a calcular  $Q_1$ :

$Q_1$	$[(\sim Q_0) \vee (Q_1 \text{ anterior})] \&\& [(Q_0) \vee (\sim Q_1 \text{ anterior})]$
$Q_1$	$[(\sim 0) \vee (1)] \&\& [(0) \vee (\sim 1)]$
$Q_1$	$[1 \vee 1] \&\& [0 \vee 0]$
$Q_1$	$(1) \&\& (0)$
$Q_1$	$(0)$

Y, con los valores de  $Q_0$  y  $Q_1$ ,  $Q_2$ :

$Q_2$	$[(\sim Q_0) \vee (Q_2 \text{ anterior})] \&\& [(\sim Q_1) \vee (Q_2 \text{ anterior})] \&\& [(Q_0) \vee (Q_1) \vee (\sim Q_2 \text{ anterior})]$
$Q_2$	$[(\sim 0) \vee (1)] \&\& [(\sim 0) \vee (1)] \&\& [(0) \vee (0) \vee (\sim 1)]$
$Q_2$	$[1 \vee 1] \&\& [1 \vee 1] \&\& [0 \vee 0 \vee 0]$
$Q_2$	$(1) \&\& (1) \&\& (0)$
$Q_2$	$(0)$

Los valores de salida son, entonces:

$$Q_0 = 0 \quad Q_1 = 0 \quad Q_2 = 0$$

Quedando demostrado que, en el caso límite en que los registros almacenen el número máximo permitido por 3 bits, los mismos, vuelve al caso base donde  $Q_0 = Q_1 = Q_2 = 0$ .

## Apéndice B

**Cuadro I:** Tabla de verdad para la compuerta lógica AND B

CLK	DIN	a	CLK && DIN && a
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

**Cuadro II:** Tabla de verdad para la compuerta lógica OR C

A	a	A    a
0	0	0
0	1	1
1	0	1
1	1	1

## Acerca de este documento

Copyright © 2020 por Roberto Nicolás Savinelli.

Distribuido bajo una licencia Creative Commons Atribución/Reconocimiento - NoComercial - CompartirIgual 4.0 Internacional (BY-NC-SA). Para más información acerca de los permisos y restricciones de la mismas considere visitar: <https://creativecommons.org/licenses/by-nc-sa/4.0/>

La información contenida en este documento fue desarrollada persiguiendo fines académicos para el curso K1001 de Arquitectura de Computadoras de UTN FRBA (Universidad Tecnológica Nacional Facultad Regional Buenos Aires) como se dictó en su formato anual en 2020 por el Ingeniero Gonzalo Vilanova.

Para más información sobre el autor visite: <https://rnsavinelli.github.io/>

Fuentes de información y/o bibliografía consultada:

STALLINGS William. Organización y Arquitectura de Computadores (Quinta Edición). PRENTICE HALL IBERIA, Madrid, 2000. ISBN: 84-205-2993-1

QUIROGA Irma Patricia, Arquitectura de Computadoras (Primera Edición). Alfaomega Grupo Editor Argentino, Buenos Aires, 2010. ISBN 978-987-1609-06-2

Aspnes James, 2020, Notes on Discrete Mathematics for the Fall 2017 semester version of the Yale course CPSC 202a, Mathematical Tools for Computer Science.

<http://www.cs.yale.edu/homes/aspnes/classes/202/notes.pdf>.