

Sequence table method for the solution of pneumatic and hydraulic circuits

Édison de Jesús Henao Castañeda, Mauricio Monroy Jaramillo

Facultad de Ingeniería Mecánica, Universidad Tecnológica de Pereira (Colombia)
edisonhenao@utp.edu.co, mauriciomonroy@utp.edu.co

Abstract

This article presents the sequence table method, used for solving motion sequences in pneumatic and hydraulic using basic skills. This method is a useful tool for teaching and for the industry because it provides a solution in an organized and systematic manner. Equations that relate inputs and outputs are obtained from the sequence table, as well as memories are used to avoid the interference of signals. The equations are simplified in order to achieve optimal results that can be implemented with basic pneumatics or electro-pneumatics. These equations allow the simulation of the design to be done with electronic sequential logic.

Keywords: hydraulic and pneumatic systems, automatism, sequential system.

Método tabla de secuencias para la solución de circuitos neumáticos e hidráulicos

Resumen

En este artículo se presenta el método tabla de secuencias, útil para solucionar secuencias de movimiento en sistemas neumáticos e hidráulicos a partir de conocimientos básicos. Este método es una herramienta útil para la docencia y para la industria debido a que permite alcanzar una solución de un modo organizado y sistemático. De la tabla de secuencias se obtienen ecuaciones que relacionan las entradas y las salidas, además, se introducen memorias que evitan la interferencia de señales. Las ecuaciones se simplifican con el fin de llegar a resultados óptimos que pueden implementarse con neumática básica o con electroneumática. Estas ecuaciones permiten que el diseño se pueda simular mediante lógica secuencial electrónica.

Palabras clave: Sistemas neumáticos e hidráulicos, automatismo, sistema secuencial.

1. Introducción

Para la solución de circuitos neumáticos o hidráulicos se tienen diversos métodos, entre ellos el de tipo intuitivo, el método de cascada, paso a paso, secuenciador, GRAFCET, entre otros [1, 2, 3].

En el método intuitivo se aplica la regla "La señal procedente del final de cada movimiento se aplica al siguiente movimiento", sin embargo, existe el riesgo de generar conflictos entre las señales que activan las válvulas que comandan los

actuadores. En este método, para circuitos simples, el problema se soluciona con relativa facilidad, pero para circuitos más complejos ya no es tan evidente una solución, esto es, se requiere de una gran habilidad. Por otro lado, se requiere confirmar lo que se intuye [2]. Por ello, se requieren métodos que permitan de una forma ordenada y sistemática alcanzar la solución.

En la literatura se encuentra información referente a los métodos en cascada y paso a paso para la solución de circuitos secuenciales. En estos métodos, se resuelve el automatismo median-

te la separación de la secuencia de movimientos en grupos, donde cada uno de estos grupos contiene movimientos de actuadores distintos [2, 3]. Para el método cascada, el número de memorias es igual al número de grupos menos uno. Para el método paso a paso, hay tantas memorias como grupos [3]. Estos métodos demandan de un nivel de conocimiento alto para su implementación, además contienen reglas que pueden olvidarse con facilidad.

El método secuenciador se suele utilizar en aplicaciones con movimientos repetitivos y utiliza módulos especiales que envían individualmente señal a grupos de válvulas en cada fase del automatismo [2].

El método GRAFCET se basa en etapas y transiciones, donde en cada etapa de una secuencia se dan determinados movimientos de los actuadores y para pasar de una etapa a otra, se deben cumplir condiciones lógicas dadas por las entradas de la aplicación. Sin embargo, GRAFCET es un método preferible para automatismos muy complejos, donde no se busca minimizar el número de estados [1, 2].

El método de tabla de secuencia está basado en conceptos de electrónica digital (lógica secuencial), a pesar de ello, como se muestra más adelante, no se requiere de conocimientos en el área. El procedimiento sistemático seguido para la solución de un circuito neumático o hidráulico se basa en pasos simples de fácil implementación. Con el procedimiento planteado se obtienen ecuaciones simplificadas que requieren de un número mínimo de componentes para su implementación. Estas ecuaciones permiten que el circuito se pueda modelar con elementos electrónicos, neumáticos, electroneumáticos, hidráulicos o electrohidráulicos.

2. Contenido

2.1. Consideraciones a partir de los sistemas digitales

Un automatismo neumático o hidráulico se puede entender como un sistema secuencial, esto es, un sistema que se caracteriza porque sus señales de salida son función de las señales de entrada actuales y condiciones pasadas (estado), es decir, es un sistema con memoria [1, 4, 5]. De

igual manera, es un sistema digital porque los valores posibles de sus señales se representan por medio de números de base binaria. Una consideración adicional para estos automatismos es que se pueden modelar como sistemas asíncronos y de tipo Moore, dado que las salidas solamente están en función de los estados actuales y no existe una señal periódica (reloj) que autorice el cambio de estados [4, 5].

Para representar el avance de una secuencia de movimiento de un automatismo hidráulico o neumático, se utiliza un gráfico denominado diagrama espacio-fase [2, 3]. Para los sistemas digitales secuenciales existen representaciones operativas como el diagrama de tiempos y la tabla de transición de estados [4, 5]. En particular, un diagrama de tiempos es similar a un diagrama espacio-fase y muestra el comportamiento de las entradas y salidas del sistema digital [2, 7]. La tabla de transición por su parte, a partir de un diagrama de tiempos, define en la forma de un cuadro los estados siguientes y las salidas correspondientes a cada uno de dichos estados, con base en las entradas actuales y los estados presentes [4, 6, 7]. Con base en lo anterior, es posible asociar el concepto de fase de un diagrama espacio-fase de un automatismo con el concepto de estado o etapa, para su análisis como sistema digital secuencial.

Con el fin de seguir una secuencia de movimientos, es necesario que el sistema a implementar posea elementos de memoria que retengan el hecho de encontrarse en alguna etapa dentro de dicha secuencia. Dichos elementos de memoria deben ser de tipo binario (biestables), acorde con la definición de sistemas digitales. Para sistemas neumáticos e hidráulicos, los biestables son válvulas direccionales de dos posiciones, las cuales deben poseer un pilotaje para cada una de las dos posiciones mencionadas. En el caso de sistemas eléctricos, se utilizan relevos con contactos enclavados. Finalmente, para los sistemas electrónicos, se utilizan elementos biestables conocidos como *latches*.

A priori, se puede afirmar que si un automatismo posee k fases o etapas, el número mínimo de elementos de memoria v se puede calcular según la ecuación algebraica 1 como:

$$v = \text{ceil}(\log_2 k) \quad (1)$$

Por ejemplo, para 5 etapas se requieren al menos $ceil(\log_2 5) = ceil(2,3219) = 3$ memorias.

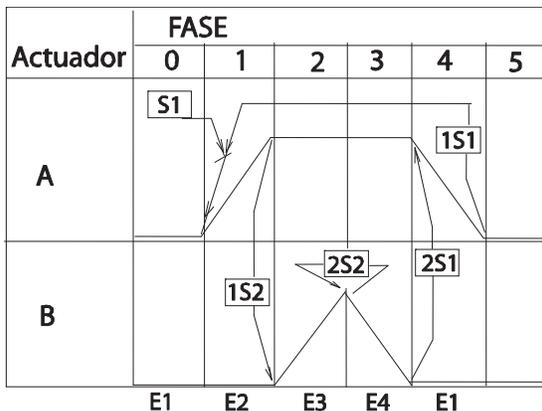
En general, para reducir el número de etapas se puede recurrir a una minimización de estados, acorde con la metodología propuesta por Hayes [4], la cual se encuentra fuera del alcance de este trabajo.

Sin embargo, un problema importante a resolver en un automatismo neumático o hidráulico consiste en evitar la simultaneidad de señales de pilotaje contrapuestas en válvulas direccionales de dos posiciones con memoria, que a su vez comandan a los actuadores [1-3]. Para resolver este problema, se puede recurrir a la intuición, a separar la secuencia en grupos, o a separar la secuencia etapa por etapa, de acuerdo con algunos de los métodos mencionados anteriormente.

Otro problema a resolver consiste en utilizar el menor número posible de elementos de entrada y elementos lógicos combinacionales. Algunos automatismos complejos pueden requerir para esto técnicas como propiedades del álgebra de Boole, métodos de Karnaugh, Quine-Mc Cluskey, etc. [5, 6].

En el método que se propone en este trabajo, se utilizan dispositivos de memoria para evitar la simultaneidad de pilotajes, así como la posibilidad de que existan condiciones de entrada iguales para distintas etapas o estados del automatismo.

Con el fin de explicar mejor el origen del método de tabla de secuencia, se parte de un ejemplo de aplicación que busca ser lo más representativo posible.



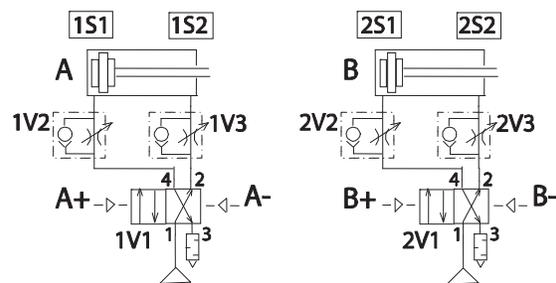
a) Diagrama espacio-fase

En la Figura 1(a) se presenta el diagrama espacio-fase de un automatismo para mover dos cilindros neumáticos de acuerdo con la secuencia {A+},{B+},{B-},{A-}.

En la parte (b) de la misma figura se ilustran los elementos neumáticos básicos para los actuadores A y B, junto con sus detectores de posiciones extremas. La válvula 1V1 comanda al actuador A y la válvula 2V1 comanda al actuador B. Las salidas de las válvulas 1V1 y 2V1 son los pórticos 2 y 4 de cada una. Dichas salidas tienen dos valores posibles de presión: compresor (1) y retorno a la atmósfera (0) y se mantienen en esos valores hasta nueva orden, es decir, 1V1 y 2V1 son elementos con memoria.

Para la solución de esta aplicación se definen:

- Entradas: S1: mando de inicio, pulsador.
 1S1: detector de posición 0 para el actuador A, esto es, indicador de actuador A retraído.
 1S2: detector de posición 1 para el actuador A, o bien, actuador A extendido.
 2S1: detector de posición 0 para el actuador B.
 2S2: detector de posición 1 para el actuador B.
- Salidas: A+: pilotaje para avance del actuador A. Al actuar, el pórtico 4 de 1V1 tendrá salida 1 y a su vez el pórtico 2 tendrá salida 0, pues 4 y 2 son mutuamente complementarios.



b) Actuadores y válvulas básicas

Figura 1. Diagrama espacio-fase y actuadores del automatismo.

A-: pilotaje para retroceso del actuador A. Al actuar, el pörtico 2 de *1V1* tendrá salida 1.

B+: pilotaje para avance del actuador B.

B-: pilotaje para retroceso del actuador B.

A partir del diagrama espacio-fase en la Figura 1(a) se puede notar que:

- Las condiciones de la fase 0 corresponden al primer estado E1, donde ambos cilindros están retraídos.
- La fase 1, denominada etapa E2, corresponde a la extensión del cilindro A, o bien, la etapa {A+}. Esta se da cuando se pulsa el mando de inicio S1 y el cilindro A está retraído, esto indicado por *1S1*.
- La fase 2, denominada etapa E3, corresponde a la extensión del cilindro B, o bien, {B+}. Esta se da cuando el cilindro A está extendido, esto indicado por *1S2*.
- La fase 3, denominada etapa E4, corresponde al retroceso del cilindro B, esto es, {B-}. Esta etapa inicia cuando el cilindro B está completamente extendido, indicado por *2S2*.
- En la fase 4, el cilindro A retrocede, esto es, {A-}. Esto es iniciado por *2S1*.

Claramente, dada la secuencia de movimientos, se ha retornado al estado E1.

La fase 5 es exactamente igual a la fase 0, por tanto es redundante.

La tabla de transición de estados (Tabla 1), acorde con [4, 6], consigna en la primera columna cada uno de los 4 estados actuales (etapas) E1 a E4. En la segunda columna se asignan los esta-

dos siguientes que se presentan ante ciertas condiciones de entradas presentes.

La columna de salidas *B* y *A* se expresa para los estados presentes E1 a E4.

Se debe entender *A=1* como extensión del cilindro A y *A=0* como retracción del mismo. Ídem para B.

Las flechas adicionales mostradas en la Tabla 1 muestran la evolución de las etapas del automatismo desde E1 a E2, E3, E4 y de regreso a E1. Todo esto por medio de las condiciones de entrada, en su orden: 11010, 00110, 00101 y 00110. El resto de estados siguientes consignados en la segunda parte de la Tabla 1 corresponden a mantenerse en la misma etapa, si no se cumple la condición de entrada específica para realizar la transición a la siguiente etapa.

Una vez se ha determinado el número de etapas, se tiene que el número mínimo de elementos de memoria debe ser 2, de acuerdo con la ecuación 1.

Por otro lado, las condiciones de entrada incluyen cinco términos: *S1*, *1S1*, *1S2*, *2S1*, *2S2*.

Sin embargo, nótese en la Tabla 1 que la condición de entrada 00110 se repite en la misma columna, esto es, para pasar de E2 a E3 y de E4 a E1 se tienen los mismos valores de entrada *S1=0*; *1S1=0*; *1S2=1*; *2S1=1*; *2S2=0*. Esto puede conducir a un conflicto de pilotajes o a una operación errónea.

El método de la tabla de secuencia se basa en la tabla de transición de estados, como la que se ha mostrado y considera únicamente los estados siguientes y sus condiciones necesarias de transición.

Tabla 1
Tabla de transición de estados

Estado presente	Entrada presente {S1,1S1,1S2,2S1,2S2}			Salidas	
	11010	00110	00101	B	A
E1	E2	E1	E1	0	0
E2	E2	E3	E2	0	1
E3	E3	E3	E4	1	1
E4	E4	E1	E4	0	1

2.2. Pasos para la formulación de la solución de una aplicación por tabla de secuencia

- Definir las entradas y salidas del automatismo. Se deben utilizar válvulas de dos posiciones para los actuadores, e incluir para cada actuador detectores de proximidad, interruptores de final de carrera o similares en ambas posiciones extremas.
- Realizar el diagrama espacio-fase del automatismo, esto es, la secuencia de movimientos.
- Formular la tabla de secuencia. En la primera columna van las salidas de pilotaje de etapa siguiente. En las siguientes columnas van los términos correspondientes a las entradas para las condiciones lógicas que llevan a dichas salidas de etapa siguiente, sin considerar pulsadores de inicio.
- Cuando se presenten condiciones de entradas repetidas en distintas etapas, se agregan a la tabla de secuencia columnas con términos que serán memorias internas, en un número determinado por la cantidad de repeticiones de la(s) condición(es) de entrada mencionada(s).
- Formular las ecuaciones lógicas para los pilotajes de cada cambio de etapa, que incluya solamente a los términos de entradas que sean de valor 1 en la tabla. Incluir los términos de memorias internas.
- Formular las ecuaciones lógicas para los pilotajes de las memorias internas. Estas ecuaciones no se simplificarán.
- Simplificar las ecuaciones lógicas de cambio de etapa, conservando los términos de entradas estrictamente necesarios para

cambiar de etapa y suprimiendo los que no son necesarios. Incluir esta vez los términos de pulsadores de inicio. En estas ecuaciones no se suprimen los términos de memorias.

- Verificar que las ecuaciones de pilotajes contrapuestos no posean los mismos términos.
- Se recomienda realizar la simulación del circuito antes de implementarlo físicamente.

Retomando el ejemplo de este trabajo, ya se han definido las entradas y salidas de la aplicación. De igual manera, ya se tiene el diagrama espacio-fase que caracteriza la secuencia de movimientos.

Se procede entonces a crear la tabla de secuencias (Tabla 2) de acuerdo al problema planteado.

En la Tabla 2 se muestran en la primera columna las etapas siguientes, esta vez definidas en forma simplificada en términos de los pilotajes para los movimientos que se han de causar. En las columnas 2 a 5 se tienen las condiciones de las señales de entrada que permiten la transición a tal etapa, sin incluir la señal del pulsador de arranque. Por ejemplo, para que se tenga la etapa A+, que corresponde al cambio E1-E2 en la Tabla 1, debe cumplirse al mismo tiempo que $1S1=1$, $1S2=0$, $2S1=1$ y $2S2=0$.

En la sexta columna se tienen señales de salida (M , \bar{M}) de tipo memoria interna (no van directamente a actuadores), pues se repiten las mismas combinaciones de señales de entrada en el caso de B+ y A-. Dado que estas memorias son binarias, se debe asignar \bar{M} a una repetición y M a la otra.

Tabla 2
Señales de entrada y salida según secuencia

		Señales de entrada				Mem
		1S1	1S2	2S1	2S2	
Señales de salida	A+	1	0	1	0	
	B+	0	1	1	0	M
	B-	0	1	0	1	
	A-	0	1	1	0	\bar{M}

Para n repeticiones, el número requerido de memorias internas m es dado por la ecuación 2 como:

$$m = \text{ceil}(\log_2 n) \quad (2)$$

A continuación, se plantean las ecuaciones lógicas 3 a 6 para las señales de salida de etapa siguiente, sin incluir la señal del pulsador de arranque $S1$.

$$(A+) = 1S1 \cdot 2S1 \quad (3)$$

$$(B+) = 1S2 \cdot 2S1 \cdot M \quad (4)$$

$$(B-) = 1S2 \cdot 2S2 \quad (5)$$

$$(A-) = 1S2 \cdot 2S1 \cdot \bar{M} \quad (6)$$

En las ecuaciones 3 a 6 van los términos que en la tabla de secuencia son de valor 1, debido a que, dada la configuración física de los detectores, no se van a presentar simultáneamente señales de indicación de extensión y retracción para un actuador dado. La operación entre términos es el producto lógico.

A continuación se escriben las ecuaciones lógicas 7 y 8 para las salidas de pilotajes para las memorias internas. Las memorias son activadas por las señales de entrada de la etapa anterior y desactivadas por las correspondientes a la etapa siguiente a su utilización. Estas ecuaciones NO deben ser simplificadas.

$$(M+) = 1S1 \cdot 2S1 \quad (7)$$

$$(M-) = 1S2 \cdot 2S2 \quad (8)$$

Cuando se dan las condiciones para el pilotaje $M+$, las salidas del elemento de memoria interna serán $M=1$ y $\bar{M} = 0$. Estos valores se invierten cuando se cumplen las condiciones de pilotaje de $M-$.

Para simplificar las ecuaciones 3 a 6, se tiene en cuenta sólo aquella señal de entrada necesaria para generar un cambio en el movimiento del actuador, las demás señales de entrada resultan redundantes y pueden ser eliminadas. Esto se puede deducir a partir del diagrama espacio-fase (Figura 1). Esta vez sí se debe incluir el

pulsador de arranque. Las ecuaciones lógicas simplificadas 9 a 12 son:

$$(A+) = 1S1 \cdot S1, \quad \text{donde } S1 \text{ es el pulsador de arranque} \quad (9)$$

$$(B+) = 1S2 \cdot M \quad (10)$$

$$(B-) = 2S2 \quad (11)$$

$$(A-) = 2S1 \cdot \bar{M} \quad (12)$$

Nótese que las ecuaciones 7 y 8 no contienen los mismos términos de entradas, lo mismo ocurre cuando se analizan las ecuaciones 9 y 12 para el actuador A y las ecuaciones 10 y 11 para B.

Por último, se procede a simular el circuito en FluidSim, Automation Studio u otro programa de simulación de circuitos neumáticos.

La solución mediante elementos neumáticos se muestra en la Figura 2. Las salidas M y \bar{M} obtienen a partir de una sola válvula 4/2 con memoria y activación neumática 3V1. Las condiciones de producto lógico de las ecuaciones 7 a 12 se realizan mediante las válvulas de simultaneidad 3V2, 3V3, 3V4, 3V5.

Para implementar la solución mediante elementos electrónicos digitales, se debe considerar que la base para el circuito secuencial es el denominado *latch RS*. Se tienen dos modelos de *latch*, uno construido a base de compuertas NAND y otro con compuertas NOR [5, 7, 8]. Las conexiones entre compuertas para el *latch RS NOR*, su símbolo y su tabla de verdad se muestran en la Figura 3.

Las salidas $Q(t+1)$ y $Q'(t+1)$ son de estado siguiente, mientras $Q(t)$ y $Q'(t)$ son valores de estado presente. Nótese que Q y Q' son salidas complementarias, salvo en el caso $R=S=1$, mas ésta última es una condición prohibida que debe evitarse.

A partir de la tabla de verdad de c) de la Figura 3 y de lo explicado en los apartados 2.1 y 2.2 acerca del funcionamiento de la válvula 4/2, es posible asociar funcionalmente a la entrada S del *latch RS* con $(M+)$ de la válvula neumática, el terminal R con $(M-)$, La salida Q con M y finalmente a Q' con \bar{M} .

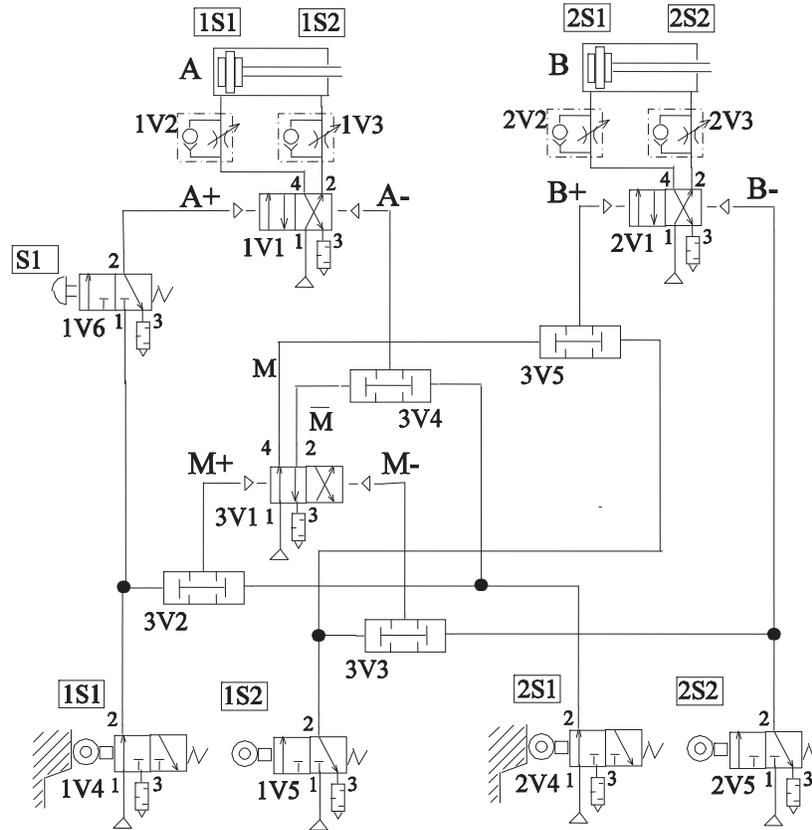
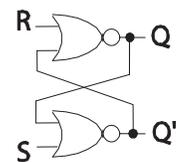
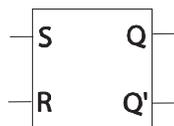


Figura 2. Secuencia A+, B+, B-, A- por medio de elementos neumáticos.



a) Esquema de compuertas



b) Símbolo

ENTRADAS		SALIDAS		ACCIÓN
R	S	Q(t+1)	Q'(t+1)	
0	0	Q(t)	Q'(t)	memoria
0	1	1	0	Set
1	0	0	1	Reset
1	1	0	0	prohib.

c) Tabla de verdad

Figura 3. Latch RS NOR y tabla de verdad [6, 7].

De este modo, las ecuaciones 13 a 18 para los terminales Set y Reset de los 3 latches de la solución electrónica, a partir de las ecuaciones 7 a 12, son:

Latch 1:
 asociado a (A+): $Set = 1S1 \cdot S1$ (13)
 (A-): $Reset = 2S1 \cdot \bar{M}$ (14)

Latch 2:
 asociado a (B+): $Set = 1S2 \cdot M$ (15)
 (B-): $Reset = 2S2$ (16)

Latch 3:
 asociado a (M+): $Set = 1S1 \cdot 2S1$ (17)
 (M-): $Reset = 1S2 \cdot 2S2$ (18)

La implementación de la solución electrónica se muestra en la Figura 4, en la cual se ilustran los *latches* 1 y 2 cuyas salidas van a comandar a los actuadores A y B, respectivamente. Para este propósito, cada una de las válvulas 1V1 y 2V1 (Figura 1) deben tener pilotaje por medio de dos solenoides, uno para cada posición. De igual manera, se ilustra el *latch* 3, análogo a la memoria M y las compuertas lógicas AND para las diferentes condiciones de producto, propias de las entradas del automatismo.

3. Resultados

El método de tabla de secuencia ha sido enseñado en cursos de neumática y electroneumática para la Maestría en Sistemas Automáticos de Producción en la Universidad Tecnológica de Pereira. Dicho método se ha formulado e implementado físicamente con éxito, para diferentes aplicaciones con 2, 3 y 4 actuadores y hasta 16 etapas.

El número de elementos totales de memoria *L* necesarios para implementar un automatismo por medio del método de tabla de secuencia se puede calcular mediante la ecuación 19 como sigue:

$$L = N + R \tag{19}$$

donde *N* es el número de actuadores y *R* es el número de memorias internas debidas a repeticiones.

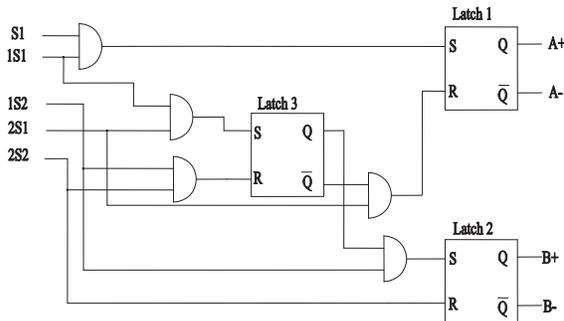


Figura 4. Diagrama esquemático de la solución.

En la Tabla 3 se indica la cantidad de elementos totales de memoria para algunas secuencias implementadas en los cursos y se comparan con otros métodos utilizados. Para el método de tabla se especifican los valores de repeticiones *R*.

El ejemplo de este trabajo podría ser resuelto con el mínimo de elementos de memoria, que es de 2, ya que hay 4 etapas. Esto es posible por medio de técnicas de minimización de estados [4] y la implementación física se realiza con componentes electrónicos, utilizando las salidas *Q* o \bar{Q} de los *latches* 1 y 2 como términos de memoria. Esto mismo no es conveniente en la solución neumática, por consideraciones de consumo de aire comprimido de los actuadores y retardo de señales.

Tabla 3
Elementos de memoria según métodos de solución

Secuencia	Tabla sec.	Método Paso a paso	GRAF CET	Método Cascada
A+, B+, B-, A- (este trabajo)	3 (R=1)	4 Grupos (Nota 1): {A+,B+}, {B-,A-}	5 (Nota 2)	3 Grupos (Nota 3): {A+,B+}, {B-,A-}
A+, A-, B+, A+, A-, B-	4 (R=2)	6 Grupos: {A+}, {A-,B+}, {A+}, {A-,B-}	7	5 Grupos: {A+}, {A-,B+}, {A+}, {A-,B-}
A+, C+, C-, B+, A-, B-	4 (R=1)	6 Grupos: {A+,C+}, {C-, B+, A-}, {B-}	7	5 Grupos: {A+,C+}, {C-, B+, A-}, {B-}

Nota 1: Para el método paso a paso, se suman actuadores y grupos, como se mencionó previamente.
 Nota 2: Al utilizar GRAFCET, se pueden tener tantos elementos de memoria como etapas presenta el automatismo más uno. Para el ejemplo de este trabajo son 5 etapas: {Posicionamiento Inicial, A+, B+, B-, A-}, de acuerdo con la formulación que proponen Balcells y Romeral [1].
 Nota 3: Para el método cascada, se suman actuadores y grupos menos uno, como se mencionó al inicio.

4. Conclusiones

Este método sirve para resolver secuencias de movimiento de actuadores en una forma organizada, evitando la interferencia entre señales, ya que las memorias permiten discriminar condiciones de finales de carrera que se repiten.

A partir de las simplificaciones, se obtienen circuitos con un número reducido de componentes, comparado con otros métodos sistemáticos (cascada, paso a paso, GRAFCET).

En este método no es necesario realizar una separación por grupos de señales, debido a que las ecuaciones se obtienen directamente de la tabla de secuencias. Esto hace más simple el desarrollo de la solución tanto neumática como electroneumática o electrónica.

Si se desea tener un mínimo de elementos de memoria, en aplicaciones con un gran número de etapas, se pueden utilizar técnicas de minimización de estados.

En circuitos digitales electrónicos, no se suelen preferir los *latches RS* para diseño de automatismos, dado el fenómeno de la incertidumbre de retardos, vista en el modelo de sistemas digitales secuenciales asíncronos de Huffmann, que puede causar problemas de carreras críticas. Sin embargo, en estas aplicaciones, utilizando este método, la velocidad y secuencia de movimiento de los elementos y la forma de cambio de

las señales hace poco probable que el fenómeno de incertidumbre de retardos se presente.

Referencias bibliográficas

1. Balcells, J. Romeral, J. Autómatas programables. 1ª ed. Alfaomega. 2002. Pp. 440.
2. Creus, Antonio. Neumática e Hidráulica. 1ª edición. Alfaomega Marcombo. 2007. Pp. 408.
3. Serrano, Antonio. Neumática. Ed. Paraninfo. 1996. Pp. 390.
4. Hayes, John. Introducción al diseño lógico digital. 2ª edición. Addison-Wesley. 1996. Pp. 940.
5. Mandado, E. Mandado, Y. Sistemas electrónicos digitales. 9ª ed. Marcombo. 2008. Pp. 883.
6. Liu, C.L. Elementos de matemáticas discretas. 2ª edición. Mc. Graw Hill. Pp. 432.
7. Floyd, Thomas. Fundamentos de electrónica digital. 1ª edición. Editorial Limusa. 2004. Pp. 811.
8. Tocci, R. Sistemas digitales. Principios y aplicaciones. 6ª ed. Pearson Educación. 1996. Pp. 833.

Recibido el 11 de Enero de 2011
En forma revisada el 7 de Mayo de 2012