

Temporizadores inmersos en máquinas de estado, para el diseño de soluciones en la automatización

José Alfredo Colín Avila, M en C.¹,
Arodí Rafael Carvallo-Domínguez, M en C.¹,

Marcela Anaid Colín Campos, M. Sc.²
José Alfredo Colín Campos³

¹Instituto Politécnico Nacional, UPIITA,

²Oakland University, USA

³Universidad Aeronáutica en Querétaro, UNAQ

jcolin@ipn.mx
acarvallo@ipn.mx

mcolincampos@oakland.edu;
10741@soyunaq.mx

Referencia de este artículo [\[1\]](#).

ABSTRACT

This document shows a way to incorporate timing functions into state machines, from which solutions to problems in manufacturing automation are developed, the way this is presented here is from the essence of what state machines are used in the search for a solution to a given automation problem, in this scenario the timing functions that are handled have no relation to the concept of time interval, which is required in the execution of instructions of a program, programmed in a programmable device, which was generated from Boolean functions obtained through a methodology, in our case the one developed by Messrs. Mealy and Moore. By including timers in state machines, they are considered both as output functions and as input signals. The latter are obtained after the time programmed in the timer has been reached, signals that are derived from the time variable managed by the timer. This form of including timers in state machines has been developed only for timers of the On Delay (ON delay), TON, TMR, RTO types, among many synonyms for names given by programmable logic controller (PLC) manufacturers, although the obtained model can also be implemented with wired logic in simple cases. The handling of the Mealy-More procedure can be complicated by the use of timers, by increasing the number of variables for the transitions that they provide, however, this document also presents a way of proceeding in this context, in order to facilitate the synthesis procedure, not to optimize the number of variables in play, but to achieve a discretization to obtain stages, to obtain the output functions that will integrate the model of the solution to a posed problem.

INTRODUCCIÓN

Las herramientas existentes para diseñar soluciones en la automatización son útiles, siempre que los modelos de solución se encuadren en un procedimiento de análisis y síntesis, como es el caso de la automatización de la manufactura, los Controladores Lógicos Programables (PLCs), siguen siendo considerados en los sistemas de control en la automatización industrial, debido a su forma de manejar el control de tipo On-Off, y más aún, porque tienen la posibilidad de usar el lenguaje de escalera (*Ladder*), para su programación; una solución que se derive de

procedimientos metodológicos, siempre será de alta confiabilidad en la operación y, óptima en tiempo y en la cantidad de elementos que la integren.

Por todo lo anterior, se ha estado trabajando en ejemplificaciones de la utilización de metodologías como es la de Mealy-Moore, para el desarrollo de modelos solución óptimos con los que se desarrollen los algoritmos para programar PLCs; aquí, se muestran los resultados de emplear esta metodología en el diseño del programa para el dispositivo programable, con el que se hará el control de la operación.

PLANTEAMIENTO

Una máquina de estado se construye con dos elementos, estados y transiciones, como se muestra en la FIGURA 1, donde q_i es el estado presente, q_{i+1} es su estado futuro, Z_i es la salida en el estado presente, Z_{i+1} es la salida en el estado siguiente y, x_i, x_{i+1}, \dots son las señales de entrada que determinan la transición del estado presente al

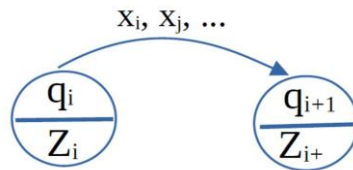


Figura 1. Modelo general de una máquina de estado.

estado siguiente; para todo estado, siempre habrá un estado siguiente o futuro si son de tipo finitas, así las transiciones sólo estarán presentes cuando una variable asignada a las entradas, se haga presente, sin embargo, las señales de entrada pueden considerar dos comportamientos más, la transición ascendente y la descendente; consideración que flexibiliza la funcionalidad de las expresiones de salida en su ejecución en un PLC, los diagramas de escalera son quizá el mejor formato a emplear para las expresiones booleanas que definen la secuencia de operación de una máquina de estado.

Conservando las restricciones que de origen rigen el desempeño de las máquinas de estado, ante la necesidad de usar temporizadores en el desarrollo de una solución, es de considerarse que esta variable tiene una parte como función de salida y origina una señal que corresponde a una entrada de una transición de una máquina de estado, se ha ideado la forma de considerar estas variables al construir la máquina de estado, considerando al temporizador como salida, su asignación se ubica dentro del estado que se le asigna T_i , y la parte que es respuesta operando como señal de entrada, cuando se ha cumplido el tiempo programado, se ubica en una transición t_i , en muchos casos, esta transición parte del estado donde se ubicó al temporizador como salida, como se muestra en la FIGURA 2.

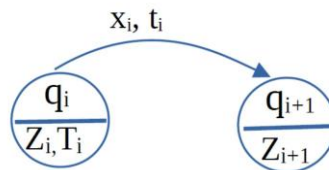


Figura 2. Máquina de estado con temporizador.

DESARROLLO

Para mostrar el manejo de los temporizadores en las máquinas de estado, se desarrolla el siguiente ejemplo.

Ejemplo 1. Se tiene la necesidad de controlar la operación de un motor eléctrico, de forma que, se controle el paro y arranque con dos botones de contacto momentáneo, A y P; cuando se pulse A inicia el ciclo, en ese momento un temporizador empieza a operar y cuando ha transcurrido el tiempo t_0 programado en éste, el motor eléctrico, al

que se le asigna la etiqueta M comienza a operar y para detenerlo se debe pulsar el botón P asignado para el paro. También, en cualquier momento que se pulse el botón P se detiene el motor M. La máquina de estado que corresponde a la problemática descrita se muestra en la FIGURA 3.

Para realizar la tabla característica, se debe conocer el número de *Flip-Flops* (FF), que se requieren para el diseño de la solución. Esto se puede obtener con la expresión $2^n \geq k$, donde n es el número de FF, mientras que, k es el número de estados de la máquina de estados. Al contar con tres estados, se deben emplear dos FF en el modelo solución; entonces, los estados resultantes se muestran en la TABLA 1, donde se aprecian las columnas asociadas al temporizador T0, como salida y la señal que emana de él cuando se alcanza el tiempo programado.

La Tabla de estado, se reformula empleado el código Gray y estados de no importa en variables de entrada, para facilitar esta etapa del diseño, dando como resultado la TABLA 2. Antes de continuar con el proceso de diseño,

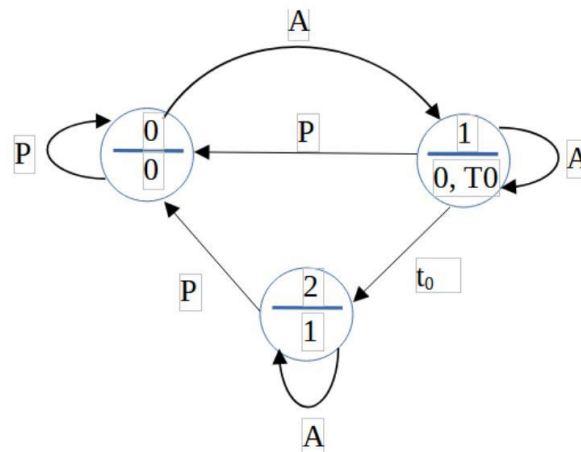


Figura 3. Máquina de estados del ejemplo1.

empleando el método de Mealy–Moore, se hace el comentario que sostiene la restricción que establece que “la duración de la señal de entrada que provoca el cambio al estado siguiente, debe tener una duración en el tiempo menor que el tiempo que tarda en cambiar el circuito del estado presente al estado siguiente” [1].

Q	P	A	t ₀	M	TIMER
0	0	1	-	0	-
1	0	1	2	0	T0
2	0	2	-	1	-
3	-	-	-	-	-
Q ^v	Q ^{v+1}		Salidas		

Tabla 1. Diagrama de estados del ejemplo 1

Q1	Q0	P	A	t ₀	M	T
0	0	00	01	XX	0	-
0	1	00	01	10	0	T0
1	1	XX	XX	XX	X	-
1	0	00	10	XX	1	-
Q ^v	Q ^{v+1}		Salidas			

Tabla 2. Tabla de estado reformulada

Con la información que se tiene en la TABLA 2 y considerando la tabla de excitación del FF S-R, que se muestra en la TABLA 3, se obtiene la tabla característica que se muestra en la TABLA 4.

Q	Q ^t	S	R
0	0	0	X
0	1	1	0
1	0	0	1

		S1			R1			S0			R0		
Q1	Q0	P	A	t ₀	P	A	t ₀	P	A	t ₀	P	A	t ₀
0	0	0	0	X	X	X	X	0	1	X	X	0	X
0	1	0	0	1	X	X	0	0	X	0	1	0	1
1	1	X	X	X	X	X	X	X	X	X	X	X	X
1	0	0	X	X	1	0	X	0	0	X	X	X	X

Tabla 3. Tabla característica del ejemplo 1

Tabla 4. Tabla de excitación de un FF S-R

La Tabla característica contiene la información, de las expresiones booleanas que compondrán el Modelo solución, en esta tabla aparece también la variable t_0 , que corresponde a la salida del temporizador $T0$, pero que en la máquina de estado se maneja como variable de entrada.

OBTENCIÓN DE LA SOLUCIÓN

Conforme a la metodología de Mealy-Moore, las expresiones booleanas que se obtienen, de la Tabla característica son las siguientes:

$$S0 = A * \overline{Q1}$$

$$R0 = A + t_0$$

$$S1 = t_0$$

$$R1 = P$$

Y para la obtener las salidas que, en este caso, es para la variable M y el temporizador $T0$, se apoya en la Tabla de estado, localizando las columnas correspondientes, donde cada variable tiene en valor de "1" y la combinación de los estados presentes que en este caso son las que se muestran a continuación:

$$M = Q1 * \overline{Q0}$$

$$T0 = \overline{Q1} * Q0$$

Con estas expresiones se puede construir el diagrama de escalera para programar un PLC, realizando la codificación adecuada a su sintaxis.

AMPLIACIÓN DE TEMPORIZADORES

Como se menciona anteriormente, cuando crece el número de temporizadores a emplear en una solución, el tratamiento para desarrollar el diseño se torna abrumador, dado que como la información se vierte en tablas, como se aprecia en el ejemplo anterior, al crecer el número de variables, crece el número de columnas y, si el número de estados aumenta, la tabla se expande aún más, lo que puede originar que, este crecimiento provoque algunas confusiones al momento de subir la información correspondiente a cada columna de cada variable de

entrada; en estos casos, la sugerencia es separar por secciones la tabla característica, una sección con columnas de las funciones SET y RESET, que correspondan a cada FF. A continuación, se muestra con un ejemplo esta solución.

Ejemplo 2.- Se trata de un sistema de control de acceso de tipo bidireccional que tiene los sensores

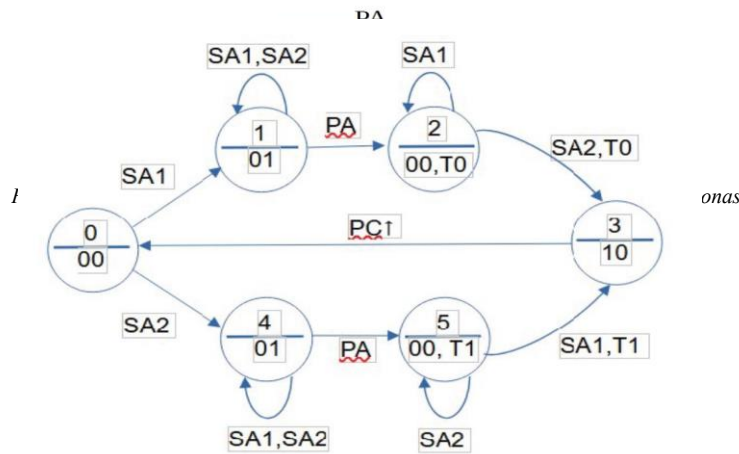


Figura 5. Máquina de estados del ejemplo 2.

necesarios para ello, ver FIGURA 4; en el caso de las salidas se consideran dos, AP y CP, para abrir y cerrar las puertas, para describir su operación se desarrolla la máquina de estado que se muestra en la FIGURA 5.

Q2	Q1	Q0	EDO	SA1	SA2	PA	PC↑	T0	T1	CP	AP	TMRs
0	0	0	0	1	4	-	-	-	-	0	0	-
0	0	1	1	1	1	2	-	-	-	0	1	-
0	1	0	2	2	3	-	-	3	-	0	0	T0
0	1	1	3	-	-	-	0	-	-	1	0	-
1	0	0	4	4	4	5	-	-	-	0	1	-
1	0	1	5	3	5	-	-	-	3	0	0	T1
1	1	0	6	-	-	-	-	-	-	-	-	-
1	1	1	7	-	-	-	-	-	-	-	-	-
Q ^v			Q ^{v+1}					Z ^v				

Tabla 5. Tabla de estados del ejemplo 2

Se aprecia que se requieren tres FF, a partir de ello, se elabora la tabla de estado, la cual se observa en la TABLA 5, en la que se muestran los estados siguientes en color verde, incluyendo para las columnas de las variables de temporización t_0 y t_1 , observe que el sensor PC tiene la etiqueta de disparo por flanco ascendente, debido a que en el estado inicial "0" éste se encuentra cerrado.

S0													S1				FF1				R1				
FF0						R0							FF2												
Q2	Q1	Q0	SA1	SA2	PA	PC↑	T0	T1	SA1	SA2	PA	PC↑	T0	T1	PA	PC↑	T0	T1	SA1	SA2	PA	PC↑	T0	T1	
0	0	0	0	1	0	X	X	X	X	0	X	X	X	X	X	X	X	X	X	X	0	X	X	X	X
1	0	0	1	X	X	0	X	X	X	0	0	1	X	X	X	X	X	X	X	X	0	X	X	X	X
3	0	1	1	X	X	X	0	X	X	X	X	X	1	X	X	X	X	X	X	X	0	X	X	X	X
2	0	1	0	0	1	X	X	1	X	X	0	X	X	0	X	X	X	X	X	X	0	X	X	X	X
6	1	1	0	X	X	X	0	X	X	X	X	X	X	X	X	X	X	X	X	X	0	X	X	X	X
7	1	1	1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0	X	X	X	X
5	1	0	1	X	X	X	X	X	0	0	X	X	X	0	X	X	X	X	X	X	0	X	X	X	X
4	1	0	0	0	0	1	X	X																	

Tabla 6. Tabla característica presentada por cada FF

La tabla característica que se muestra en TABLA 6, se presenta en secciones por cada FF requerido, en ella se aprecia, los estados presentes que le corresponden a cada FF se encuentran de color verde, así el estado presente para el FF0 es $Q0$ y así sucesivamente. Para verter la información en cada una de las secciones de la tabla característica, se usa la tabla de excitación del FF S-R definida en la TABLA 3. Finalmente, se obtienen las expresiones de correspondientes a las entradas S y R de cada FF, dando como resultado:

$$\begin{aligned}
 S0 &= SA1 * \overline{Q2} * \overline{Q1} + SA2 * Q1 * \overline{Q0} + PA * \overline{Q1} * Q0 + T_0 & R0 &= PA * \overline{Q2} * Q1 + PC \uparrow \\
 S1 &= SA1 * Q2 * \overline{Q1} * Q0 + PA * \overline{Q2} + t_1 & R1 &= PC \uparrow \\
 S2 &= SA2 * \overline{Q1} * \overline{Q0} & R2 &= SA1 * \overline{Q1} * Q0 + PC \uparrow
 \end{aligned}$$

Las expresiones para las salidas del sistema CP, AP y los temporizadores $T0$ y $T1$, se obtienen directamente del número de estado en el que se encuentran ubicados, esto con las combinaciones de $Q2$, $Q1$ y $Q0$, obteniendo las siguientes expresiones:

$$\begin{aligned}
 CP &= \overline{Q2} * Q1 * Q0 & T0 &= \overline{Q2} * Q1 * Q0 \\
 AP &= \overline{Q2} * \overline{Q1} * Q0 + Q2 * \overline{Q1} * \overline{Q0} & T1 &= Q2 * \overline{Q1} * Q0
 \end{aligned}$$

Estas expresiones se programaron en dos PLCs de marcas que se encuentran tanto en el mercado como en las plantas industriales, se hizo la simulación y siempre se ejecutó la secuencia correctamente.

CONCLUSIONES

Siempre que se empleen técnicas y metodologías para realizar diseños, se obtendrán soluciones altamente confiables, lográndose una operación óptima y eficiente de éstas. Al emplear herramientas de diseño disponibles en esta área, se recomienda fragmentar la necesidad planteada en "sub-necesidades" a fin de diseñar una solución para la parte medular y, de ahí, construir alrededor de ella el complemento de soluciones requeridas para las "sub-necesidades" e integrar una la solución total. [2]. Para lo expuesto en este documento, emplear técnicas y metodologías de diseño como las mencionadas, proporciona un escenario en el que no hay necesidad de considerar la tecnología a emplear para construir la solución e implementarla, se trata de un diseño que puede implementarse con las tecnologías que sean propias de este tipo de necesidades.

Las máquinas de estado son un punto de análisis idóneo, cuando se presenta alguna inconsistencia en la solución implementada, porque se puede ubicar con agilidad y rapidez la inconsistencia que se presente y de ahí hacer la actualización requerida.

Referencias bibliográficas

F. J. Hill y G. R. Peterson, Teoría de conmutación y diseño lógico, Ciudad de México: LIMUSA, 1980.

C. Canto, «InfoPLC,» 2024. [En línea]. Available:

https://www.infoplcn.net/files/documentation/automatas/infoPLC_net_5_EL_SCAN.pdf. [Último acceso: Enero 2025].

P. Melore, «PLCS,» [En línea]. Available: www.plcs.net. [Último acceso: 28 Enero 2025].

J. A. Colín Ávila, A. I. Cruz Ramírez, H. A. Guerra Calva y J. Fonseca-Campos, «Acondicionamiento de señales de entrada para un PLC, usando la función disparo por flanco, para ser empleadas en el diseño de programas empleando circuitos de MEALY y MOORE,» Boletín UPIITA, n° 107, 2025.

J. A. Colín Ávila, Y. I. Sánchez Herrera y C. M. Chávez Garrido, «Algoritmo para un Controlador Lógico programable que opera un Dispositivo variador de Frecuencia empleando Diseño Digital,» Boletín UPIITA, n° 46, 2015.

Referencia del artículo

Colín, J., Colín. M., Carvallo-Domínguez, A. & Colín, J. Temporizadores inmersos en máquinas de estado, para el diseño de soluciones en la automatización. (*septiembre-octubre, 2025*). *Boletín UPIITA. año 20, (110) 2025.*

<https://www.boletin.upiita.ipn.mx/index.php/ciencia/1098-cyt-numero-110/2444-temporizadores-inmersos-en-maquinas-de-estado-para-el-diseno-de-soluciones-en-la-automatizacion>