

Uso de las cartas ASM

M. Aguilar Ibarra, C. Chávez Hernández, E. Díaz Hernández, D. Moreno Martínez

Departamento de ingeniería en sistemas computacionales, ESCOM IPN

jaguilar1800@alumno.ipn.mx, cchavez1900@alumno.ipn.mx, ediaz1900@alumno.ipn.mx, dmoren2002@alumno.ipn.mx

Resumen— El uso de diagramas de estado, y cartas ASM ayuda a diseñar sistemas digitales cada vez más complejos. Los diagramas de flujo se ocupan para visualizar de una mejor manera los pasos a seguir en un orden específico para un algoritmo ya sea de software o hardware. Los diagramas de máquinas de estado también conocidos como cartas ASM son un tipo de diagrama de flujo que sirve para visualizar y poder realizar algoritmos en hardware digital. La principal diferencia entre un diagrama de flujo y las cartas ASM, es que, para estas últimas, hay casos en donde se requiere de un pulso de reloj para cambiar de estado, esto quiere decir que son síncronas, y en otros casos se cambia de estado cuando se cumplen ciertas condiciones. Estas cartas en la práctica sirven para visualizar circuitos digitales complejos como lo son los FPGA, memorias RAM y ROM, microcontroladores y microprocesadores, entre otros. Aplicar de manera correcta estos diseños es importante para entender el funcionamiento de circuitos complejos y que se puedan realizar de manera correcta.

Palabras Clave —Cartas ASM, Diagramas de estado, Diagramas de flujo.

Abstract— The use of state diagrams and ASM charts helps design increasingly complex digital systems. Flowcharts are used to better visualize the steps to follow in a specific order for an algorithm, whether it be software or hardware. State machine diagrams, also known as ASM charts, are a type of flowchart used to visualize and implement algorithms in digital hardware. The main difference between a flowchart and ASM charts is that for the latter, there are cases where a clock pulse is required to change state, meaning they are synchronous, and in other cases, the state changes when certain conditions are met. These charts are useful for visualizing complex digital circuits such as FPGAs, RAM and ROM memories, microcontrollers, and microprocessors, among others. Correctly applying these designs is important to understand the functioning of complex circuits and to ensure they are implemented correctly.

Keywords -- ASM charts, State diagrams, Flowcharts.

I. INTRODUCCIÓN

El presente documento aborda el análisis de los conceptos relacionados con el empleo de diagramas de flujo y diagramas de máquinas de estado (cartas ASM). Durante el curso de "Diseño de Sistemas Digitales", se han abordado estos temas de manera teórica y práctica, lo que permite ahora realizar un estudio más exhaustivo de ellos como también poder diseñarlos.

A. Diagrama de flujo

Un diagrama de flujo es una forma conveniente de especificar la secuencia de los pasos de procedimiento y las trayectorias de decisión de un algoritmo [Tabla 1]. Un diagrama de flujo de un algoritmo de hardware se encarga de transformar las instrucciones verbales en un diagrama de información que enumera la secuencia de las operaciones junto con las

condiciones necesarias para su correcta ejecución. [1]

Diagrama de flujo y sus principales símbolos






SÍMBOLO	NOMBRE	FUNCIÓN
	Inicio / final	Representa el inicio y el final de un proceso
	Línea de flujo	Indica el orden de la ejecución de las operaciones
	Proceso	Representa cualquier tipo de operación
	Decisión	Permite analizar una situación según si su respuesta es verdadera o falsa, sí o no
	Entrada / salida	Representa la lectura de datos en la entrada y la impresión

Tabla 1 Simbología de un diagrama de flujo. [2]

Un ejemplo de diagrama de flujo es como se muestra en la Ilustración 1:

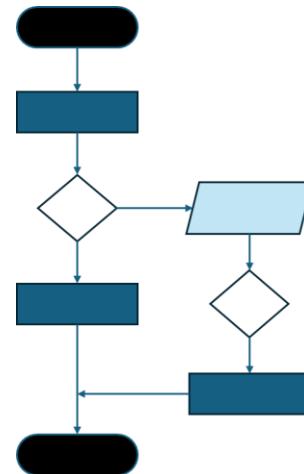


Ilustración 1 ejemplo de diagrama de flujo

B. Cartas ASM

Las cartas ASM o diagramas ASM, son diagramas de flujo que sirven para definir algoritmos específicos y ejecutarlos con hardware digital. También sirven para mostrar de forma gráfica el algoritmo de una máquina de estados. Son muy similares a un diagrama de flujo, pero con la diferencia de que hay algunos que cambian con un pulso de reloj que hace que cambien de estados, contrario a los diagramas de flujo que cambian de proceso cuando se termina la tarea correspondiente. [3]

Estos diagramas se componen de tres elementos básicos: el

cuadro de estado, el cuadro de decisión y el cuadro de condicional [Ilustración 2]. Estos cuadros se conectan con flechas o líneas dirigidos hacia el siguiente estado. [1]

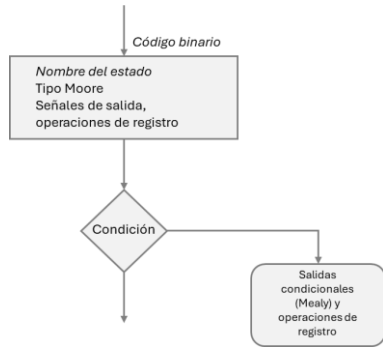


Ilustración 2 Ejemplo de diagrama básico ASM

Existen diversas formas de poner información en un diagrama ASM, una de ellas consiste en poner un cuadro de estado para indicar la secuencia de control de un estado, como se muestra en la Ilustración 3 (a). [1]



Ilustración 3 Ejemplo de cuadro de estado

En los cuadros de estado rectangulares es donde se ponen las operaciones de registro o los nombres de las señales de salida que el control genera mientras está en el estado indicado. Al estado se le debe dar un nombre identificable y se coloca en la parte superior del cuadro. El código binario asignado se coloca en la esquina superior derecha. En la Ilustración 3 (b) se muestra un ejemplo de cómo se realiza este tipo de cuadro de estado. [1]

C. Ejemplo de cartas ASM

Un ejemplo de carta ASM se puede apreciar en la Ilustración 4:

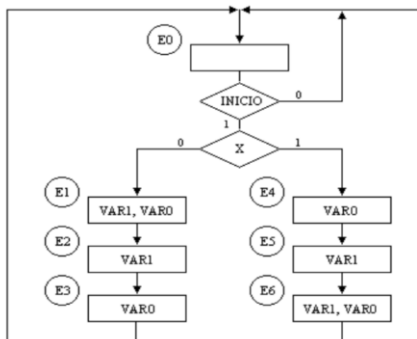


Ilustración 4 Ejemplo de carta ASM. [3]

En donde cada “E” representa un estado, siendo E0 el estado inicial, E1 el estado siguiente, luego le sigue E2 y así continua hasta llegar a E6.

El diagrama anterior sirve para representar un caso donde se tiene un dispositivo que genere cierta secuencia binaria sólo cuando la variable INICIO sea igual a uno. Junto a esto, la secuencia depende del valor de entrada “X”.

Cuando el valor de entrada “X” vale 0, se obtiene la siguiente secuencia: 11, 10, 01.

Cuando el valor de entrada es “X” = 1, se obtiene: 01, 10, 11.

Este ejemplo sirve para visualizar cómo funcionan estos algoritmos de una manera más estructurada.

II. DESARROLLO

Diseño de cartas ASM para para mostrar el alfabeto y números en un LCD de la tarjeta DE2-115 mediante un teclado PS/2

Para poder utilizar el teclado PS/2 desde la tarjeta DE2-115 se diseñó un código VHDL elaborado por un usuario llamado “Scott_1767” el cual se encuentra disponible en la página Digikey [4]. En esta página se va explicando para que sirve cada parte de los códigos proporcionados. En la introducción se menciona que se detalla como un componente de interfaz recibe transacciones de datos de un teclado PS/2 y proporciona los códigos de activación y desactivación del teclado a través de una interfaz paralela. [4]

A. Funcionamiento del teclado PS/2

Para entender cómo es que una tarjeta DE2-115 reconoce la señal de un teclado PS/2, primero hay que entender cómo funciona el teclado.

El teclado funciona mediante un conector MINI – DIN de 6 pines. La computadora debe administrarle al teclado una fuente de 5v para poder funcionar. La comunicación ocurre a través de una interfaz serial de 2 hilos, que consta de una línea de reloj y una línea de datos. Ambas líneas requieren resistencias pull-up de 2kΩ. Se necesitan resistencias en serie de 120Ω para conectar las E/S del FPGA de 3.3V a las señales de 5V. [4]

El teclado detecta cada byte recibido y lo interpreta como un código de activación (cuando se presiona una tecla) o un código de desactivación (cuando se suelta una tecla). Existen tres conjuntos diferentes de códigos de escaneo, sin embargo, la gran mayoría de los teclados usan un conjunto de códigos de escaneo de tipo PS/2 Set 2 Make, el cual se encuentra en el apéndice de la página DigiKey [4]. Un código de activación generalmente consiste en uno o dos bytes. Si un código de activación usa dos bytes, el primer byte es x"E0". El código de desactivación de una tecla dada es típicamente el mismo que su código de activación, excepto que los códigos de desactivación incluyen un byte adicional x"F0" como penúltimo byte.

Cuando se presiona una tecla, suceden una serie de cambios de

estado como los que se muestran en la Ilustración 5:

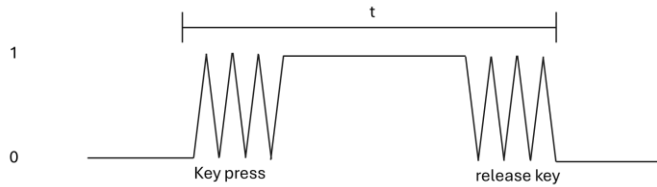


Ilustración 5 Cambios de estado al presionar una tecla en el teclado PS/2

En esta ilustración, se pueden apreciar varias cosas, la primera es que al inicio se encuentra en un estado inactivo, y al presionar la tecla cambia de estado, pero no una, sino varias veces, a este cambio se le conoce como “rebote”. [5]

El primer rebote ocurre en el momento exacto en que se presiona una tecla, este rebote depende mucho del estado físico del teclado, pero, por lo general dura entre 5 y 20 ms. [5]. En la Ilustración 6 también se puede apreciar que hay una línea recta que se encuentra en estado alto, esta línea representa el tiempo transcurrido hasta que se soltó la tecla, para finalmente llegar al segundo rebote que ocurre cuando se suelta la tecla. Pasando esto, regresa a estado bajo.

B. Interfaz del teclado PS/2

Para la interfaz del teclado inicialmente tenemos lo que se muestra en la ilustración 6:

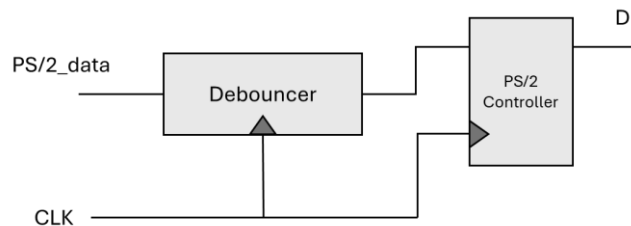


Ilustración 6 Controlador del PS/2 que recibe la señal de rebote

En esta ilustración, se tiene lo que es un “debouncer”, el cual es un circuito o algoritmo utilizado para asegurar que las señales de entrada no contengan múltiples transiciones no deseadas, que pueden ocurrir debido a la naturaleza mecánica de los botones o interruptores. [6]

Como se vio en la Ilustración 6, los rebotes que surgen al presionar o soltar la tecla pueden generar señales no deseadas, por lo que, es necesario poner un debouncer que “limpie” estas señales para que al teclado solo le pueda llegar la señal de la tecla cuando fue presionada o soltada. Después de pasar por el debouncer, la señal se dirige al PS/2 Controller, al cual también le llega una señal de reloj. Este es el componente que interpreta las señales provenientes de dispositivos PS/2, como teclados o ratones, y las convierte en datos que pueden ser utilizados por el sistema. [4] Esto se puede apreciar de una mejor manera en la Ilustración 7:

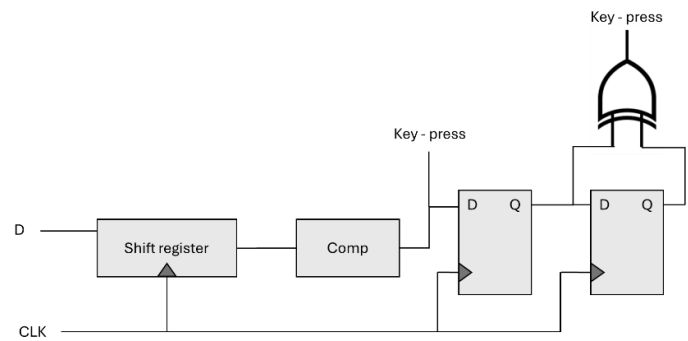


Ilustración 7 Ejemplo de carga de datos con dos flip flops de tipo D

En esta ilustración se puede observar que entra una señal a “Shift register”, el cual es un arreglo que va recibiendo una señal de ceros y unos. Luego de ahí, pasa a un comparador, el cual verifica que todas señales de Shift register se encuentren en estado alto, pero, para esto, las señales van llegando conforme llega una señal del ciclo de reloj. Entonces, al cambiar la señal del ciclo de reloj, una nueva señal va llegando al arreglo, el cual va recorriendo los valores sustituyendo los anteriores. Si, por ejemplo, el teclado se encuentra inactivo, la entrada del arreglo se vería de la siguiente forma [Tabla 2]:

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

Tabla 2 Entrada del teclado PS/2 inactivo

Si ahora se presiona el teclado, tomando en cuenta los rebotes se podrían tener las siguientes entradas [Tabla 3]:

0	1	0	1	0	1	0	1
---	---	---	---	---	---	---	---

Tabla 3 Entrada del teclado PS/2 al momento de presionar la tecla

No necesariamente se van a tener en ese orden, es solo una forma de ver los cambios de estado alto y bajo en las señales que van llegando.

Ahora, cuando se encuentra en estado alto se espera que todas las entradas se encuentren de la siguiente forma [Tabla 4]:

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

Tabla 4 Entrada del teclado PS/2 mientras se encuentra presionada la tecla

Cuando finalmente todos los valores que están en el arreglo pasan a estado alto, lo que hace el comparador es verificar que en efecto estén en estado alto y luego manda una señal de que la tecla fue presionada y también a un flip flop de tipo D, que está conectado a otro flip flop de tipo D; esto se hace con la finalidad de que al mandarse a una compuerta XOR se mande una señal en estado bajo si ambas señales de los flip flops son iguales, y en caso de ser diferentes mandaría una señal en estado alto, lo cual significaría un cambio de tecla que fue presionada o soltada.

C. Cartas ASM para mostrar el código hexadecimal en un display de 7 segmentos y en leds

En cuanto a las cartas ASM se consideró todo el funcionamiento de la interfaz del teclado PS/2 más aparte el funcionamiento para que

se enciendan los leds, mostrar los caracteres ingresados en los displays de siete segmentos, y mostrar el resultado en la LCD de la tarjeta DE2-115.

Las cartas para el funcionamiento de la interfaz del teclado PS/2 son las que se muestran en las Ilustraciones 8 y 9:

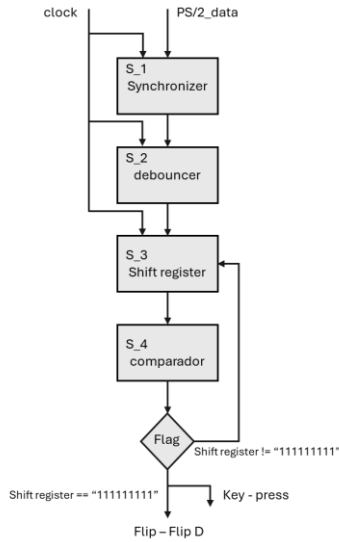


Ilustración 8 Cartas ASM de Interfaz de teclado PS/2 (a)

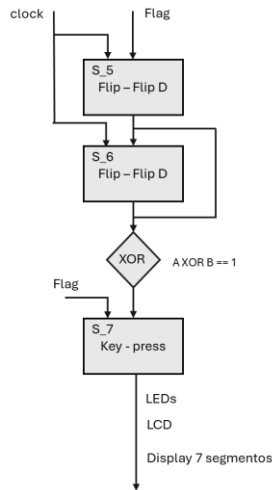


Ilustración 9 Cartas ASM de Interfaz de teclado PS/2 (b)

En cuanto a las cartas del funcionamiento de encendido de LEDs se muestran en la Ilustración 10:

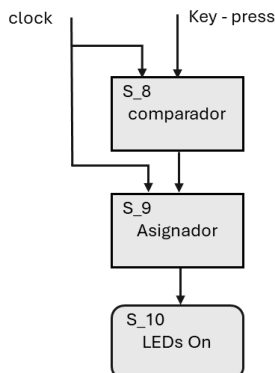


Ilustración 10 Cartas ASM de encendido de LEDs

El funcionamiento que se muestra en las cartas es que al recibir los bits de Shift Register (Ilustración 8 y 9) y luego haber pasado por la compuerta XOR, lo que hace es que compara los valores recibidos con los LEDs en donde se deben mostrar, para posteriormente mandar la asignación de LEDs a sus salidas correspondientes, dando como resultado que los LEDs se enciendan en el orden correcto.

En cuanto al funcionamiento del encendido de displays de siete segmentos se muestra en la ilustración 12:

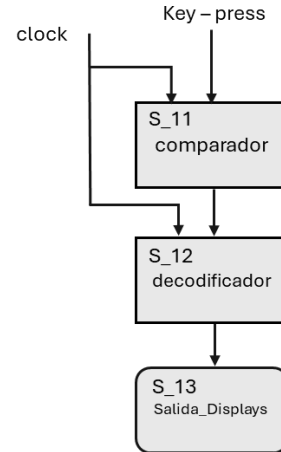


Ilustración 11 Cartas ASM de encendido de Displays de 7 segmentos

Para que se enciendan en el orden correcto los displays de siete segmentos, primero se pasa por un comparador que recibe los bits del arreglo de Shift Register (Ilustración 8 y 9) y luego haber pasado por la compuerta XOR el cual verifica que la tecla fue presionada, manda los valores al decodificador en donde dependiendo los bits recibidos los va asignando a las compuertas de los displays en un orden específico. Para esto se ocupan ciclos de reloj, en donde Clock_divider lo que hace es mandar una versión dividida del reloj después de pasar por un divisor de frecuencia.

En cuanto a las cartas para mostrar los caracteres en el LCD se muestran en la ilustración 12:

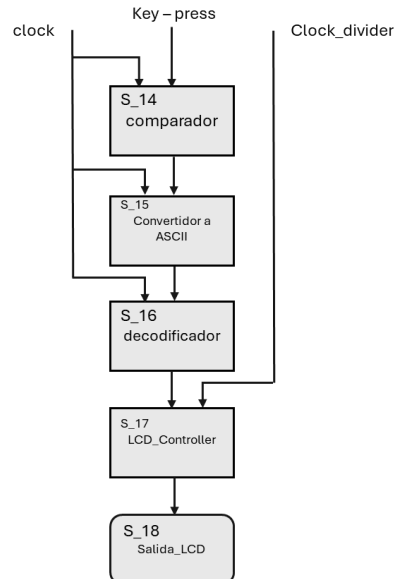


Ilustración 12 Cartas ASM para mostrar resultados en la LCD

Las cartas ASM del LCD_Controller se muestran en las Ilustraciones 13 y 14:

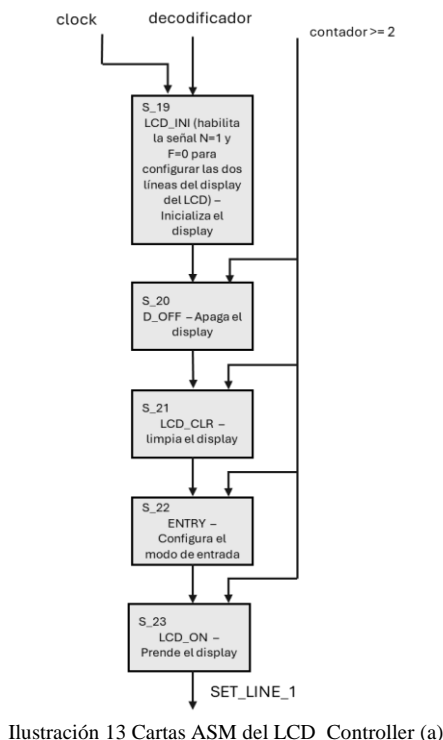


Ilustración 13 Cartas ASM del LCD Controller (a)

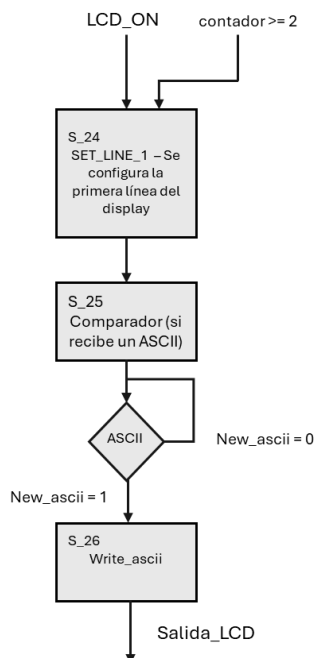


Ilustración 14 Cartas ASM del LCD Controller (b)

Para que se muestren los caracteres en la LCD, primero se pasa por un comparador que recibe los bits del arreglo de Shift Register (Ilustración 8 y 9) y luego haber pasado por la compuerta XOR el cual verifica que la tecla fue presionada, manda los valores al convertidor de binario a código ASCII, para después pasar al decodificador en donde dependiendo los bits recibidos los va mandando al LCD_Controller (Ilustraciones 13 y 14), el cual se encarga de mostrarlos en la LCD mediante un divisor de frecuencia que viene de Clock_divider, esto para que se muestren de manera correcta los resultados.

III. RESULTADOS Y DISCUSIÓN

Las salidas fueron las esperadas en la LCD, LEDs y display de siete segmentos.

IV. CONCLUSIONES

Los resultados obtenidos en la práctica cumplen con el objetivo establecido. Al explorar los diagramas de flujo, se profundizó en el concepto de las cartas ASM, lo cual facilitó la creación de un diagrama que mejoró la visualización del proceso para la recepción de caracteres desde un teclado PS/2. Este diagrama no solo permitió la visualización inicial en LEDs, sino también la posterior presentación en displays de siete segmentos y en una pantalla LCD.

AGRADECIMIENTOS (Heading 5)

Agradecemos a “[Scott_1767](#)” por haber realizado un código tan complejo que nos ayudó bastante a realizar nuestra práctica de una manera más optima y eficiente. También quisiéramos agradecer a la página “[Digikey](#)” por poner a disposición de todo publico la posibilidad de visualizar información sobre la interfaz del teclado PS/2 y su arquitectura. La documentación y los recursos ofrecidos han sido de gran ayuda para comprender y desarrollar nuestra práctica.

REFERENCIAS

- [1] Mano, M. M., & Ciletti, M. D. (2013). Diseño Digital (5.^a ed., pp. 363 - 381). Pearson Educación.
- [2] C. Giani, “Diagrama de flujo - Qué es, tipos, simbología y ejemplos - Concepto.” [Online]. Recuperado de: <https://concepto.de/diagrama-de-flujo/>
- [3] M. I. Norma Elva Chávez Rodríguez. Cartas, A. S. M. (s/f). Unam.mx. Recuperado de http://profesores.fi-b.unam.mx/normaelva/carta_ASM.pdf
- [4] [1] Scott_1767, “PS/2 Keyboard Interface (VHDL) - Logic Design - Electronic Component and Engineering Solution Forum - TechForum | DigiKey.” [Online]. Recuperado de: <https://forum.digkey.com/t/ps-2-keyboard-interface-vhdl/12614>
- [5] [1] Ing. Bruno M. Palacios, “Pulsadores y Teclados | Descargar gratis PDF | Hardware de la computadora | Algoritmos - Scribd.” [Online]. Recuperado de: <https://es.scribd.com/document/646754326/PPT-Pulsadores-y-Teclados>
- [6] [1] Leandro Alegsa, “Definición de Teclado de contacto - Alegsa.com.ar.” [Online]. Recuperado de: https://www.alegsa.com.ar/Dic/teclado_de_contacto.php#gsc.tab=0