



El teclado del computador Easy 8

Apellidos, nombre	Martí Campoy, Antonio (amarti@disca.upv.es)
Departamento	Informàtica de Sistemes i Computadors (DISCA)
Centro	Escola Tècnica Superior d'Enginyeria Informàtica

1 Resumen de las ideas clave

En este artículo vamos a describir el funcionamiento de uno de los periféricos incluidos en el computador didáctico Easy 8. Este periférico es el teclado hexadecimal, que permite introducir datos desde un teclado hexadecimal. Para usar los periféricos del EASY 8 utilizamos los puertos de entrada/salida (E/S) que podemos leer y escribir usando, respectivamente, las instrucciones IN PUERTO y OUT PUERTO. Para poder seguir este artículo docente es necesario contar con los conocimientos previos que se presentan en la Tabla 1.

Conocimientos previos
1. Instrucciones IN y OUT para acceder a la E/S.
2. Instrucciones de movimiento de datos.
3. Instrucciones aritméticas.
4. Instrucciones de comparación.
5. Instrucciones de salto.
6. Uso de etiquetas.
7. Uso básico del simulador del computador Easy 8.

Tabla 1. Conocimientos previos.

2 Objetivos

Una vez que termines de trabajar este artículo docente serás capaz de:

- Escribir un programa en lenguaje ensamblador del Easy 8 que lea datos del teclado hexadecimal y los use para cálculos sencillos.
- Ejecutar el programa en el simulador y verificar su correcto funcionamiento.

3 Introducción

El computador Easy8 es un sencillo computador didáctico construido alrededor del procesador de 8 bits Easy 8. Como cualquier computador, además de unidad central de proceso, (UCP o CPU) el computador Easy 8 está compuesto de una memoria principal de 256 bytes (256 palabras de 8 bits cada una), y un sistema de entrada/salida formado por 256 puertos de 8 bits los cuáles se acceden con instrucciones específicas, diferenciadas de las instrucciones utilizadas para acceder a memoria principal. La Imagen 1 muestra un diagrama de bloques de la estructura del computador Easy 8 donde se incluyen los periféricos disponibles en el simulador.

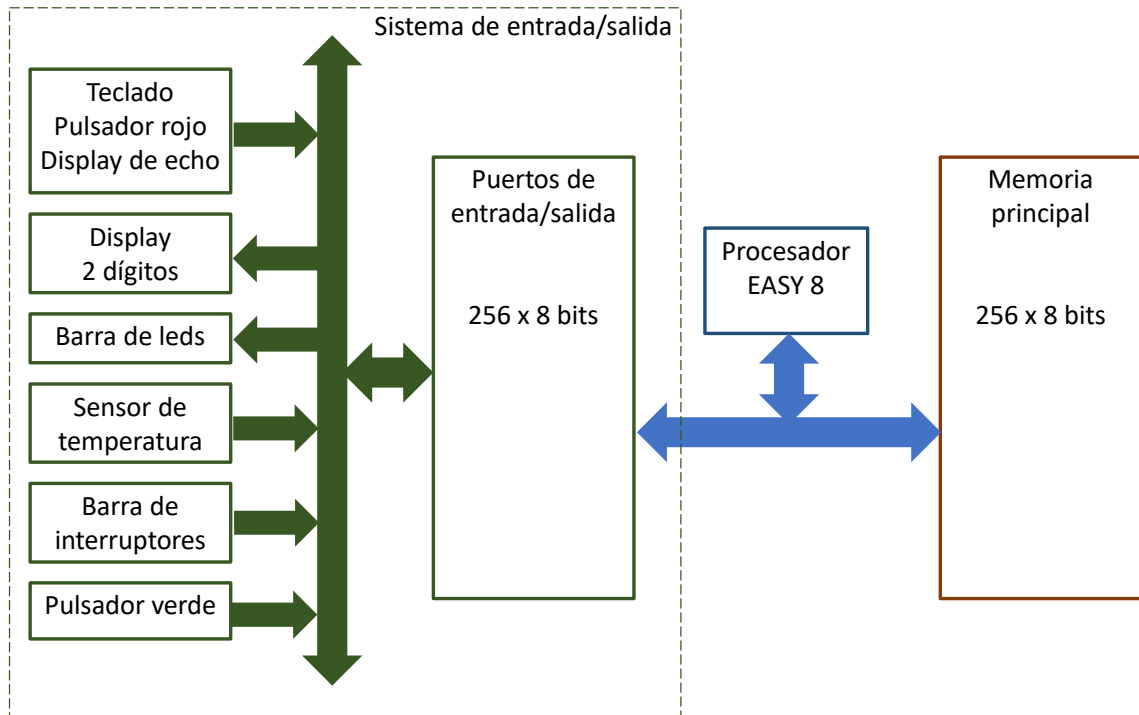


Imagen 1. Diagrama de bloques del computador Easy 8.

La Imagen 2 muestra los periféricos del Easy 8 tal como se ven en el simulador. Puedes acceder al simulador en easy8.webs.upv.es Hay que destacar el teclado hexadecimal que está compuesto por tres elementos, y que se describen con detalle en la sección siguiente.

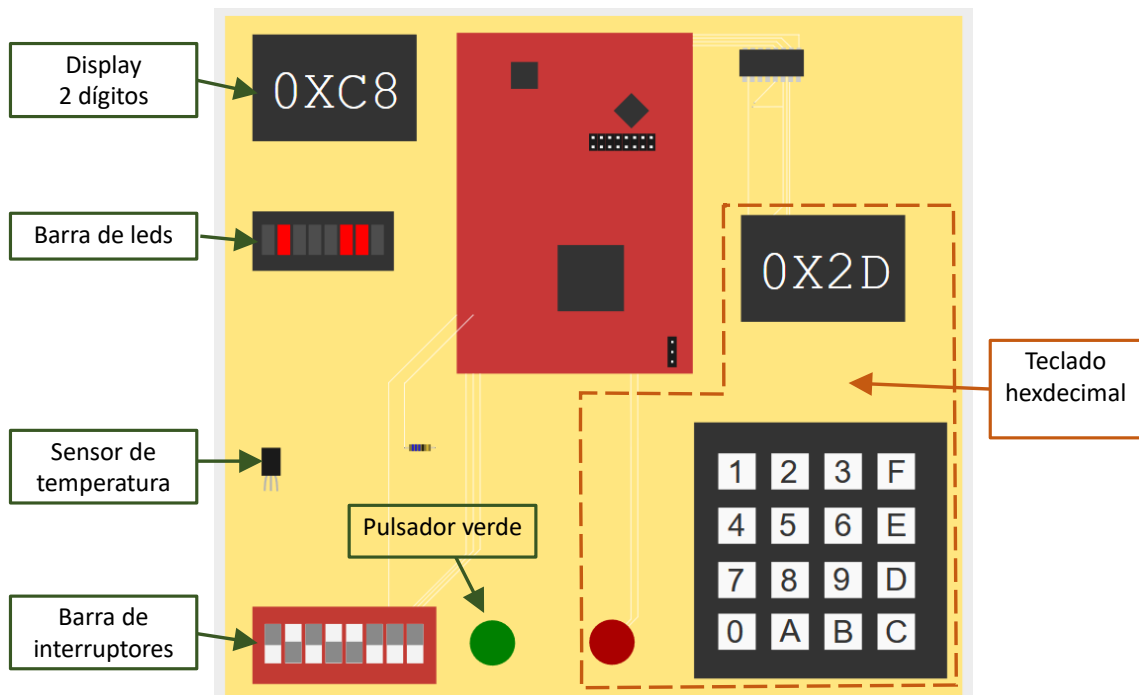


Imagen 2. Vista de periféricos del simulador del computador Easy 8.

4 El teclado del Easy 8

El teclado hexadecimal del Easy 8 es un periférico de entrada. Permite introducir un dato de 8 bits que se copia al registro RA mediante la ejecución de la instrucción de ensamblador IN. Está compuesto de tres elementos:

- Teclado hexadecimal de 16 teclas. La salida del teclado está conectada al puerto de entrada/salida (puerto de E/S) número 0 o 0x00 en hexadecimal.
- Botón rojo. Este botón está conectado al puerto de entrada/salida 2 (0x02).
- Display de eco de 2 dígitos. Su propósito es mostrar al usuario las teclas que se están pulsando en el teclado. Es importante remarcar que no está conectado a ningún puerto de entrada/salida. Es decir, su funcionamiento es autónomo y no puede ser accedido desde el procesador, y por tanto tampoco se puede acceder desde el programa que se está ejecutando.

4.1 Funcionamiento de los elementos del teclado

Es importante recordar que un dígito hexadecimal se representa por cuatro bits. La Tabla 2 muestra la correspondencia entre los dígitos hexadecimales y el código binario correspondiente.

Hexadecimal	Binario	Hexadecimal	Binario
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

Tabla 2. Equivalencia entre dígitos hexadecimales y secuencias binarias.

El teclado hexadecimal es un sistema que, si bien no es complejo, tiene un funcionamiento particular que debemos conocer para poder usarlo. Cada uno de los elementos que lo integran: el teclado, el display de eco y el pulsador rojo, juegan un papel concreto.

Cuando el usuario pulsa una tecla, el contenido del puerto de E/S 0x00 se desplaza cuatro posiciones a la izquierda (hacia el bit de mayor peso) y los cuatro bits de menor peso son sobrescritos con el código binario de la tecla pulsada. Cada vez que se pulsa una tecla se repite este proceso.

Al mismo tiempo que se pulsán las teclas, el display de eco muestra, de forma automática, el contenido del puerto de entrada 0x00. De esta forma el usuario puede comprobar la tecla que ha pulsado. La Imagen 3 muestra la evolución del contenido del puerto 0x00 y del display de eco para una secuencia de dos pulsaciones, suponiendo que el contenido inicial del puerto 0x00 es 0xA3 y que el usuario quiere introducir por teclado el valor 0x6C.

Si el usuario se equivoca al pulsar una tecla, sólo debe repetir el proceso introduciendo nuevamente los dos dígitos hexadecimales del valor deseado.

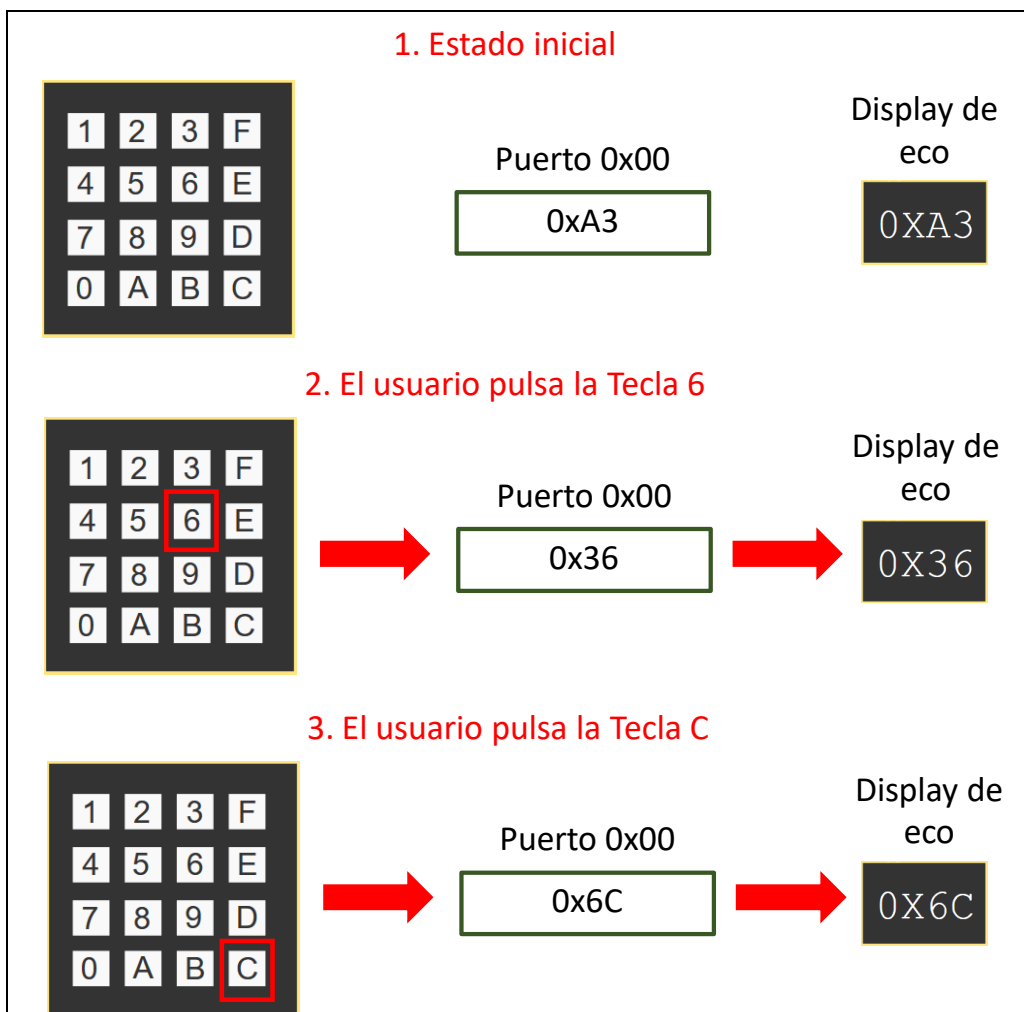


Imagen 3. Evolución del puerto de E/S 0x00 y del display de eco en una secuencia de dos pulsaciones.

Describimos ahora el comportamiento del pulsador rojo, el tercer elemento que integra el teclado hexadecimal del computador Easy 8. Cuando el usuario pulsa el botón rojo se escribe el valor 0x01 en el puerto de E/S 0x02. Si se deja de presionar el pulsador, el 0x01 escrito en el puerto de E/S 0x02 se mantiene, y se mantendrá aunque el pulsador se presione repetidamente. Es decir, se recuerda que el botón ha sido pulsado.

Cuando el programa ejecuta la instrucción IN 0x00, es decir, se lee del puerto de E/S donde está almacenado el dato introducido en el teclado, el contenido del puerto 0x02 será puesto a 0x00, borrando el indicador de que el botón rojo había sido pulsado. De esta forma el teclado queda preparado para la introducción de un nuevo dato.

4.2 Lectura del dato introducido en el teclado

En principio parece sencillo leer el dato del teclado. Simplemente sería suficiente ejecutar la instrucción IN 0x00, que copiaría el contenido del puerto de E/S 0x00 al registro RA. Sin embargo, esta instrucción, al igual que todas las demás, se ejecuta sin pausa ni espera, es decir, lee directamente el contenido del puerto independientemente de si el usuario ha tecleado el dato o no.

Por tanto es necesario establecer algún mecanismo que permita esperar a que el usuario teclee el dato antes de leerlo. Para este mecanismo se utiliza el pulsador rojo.

El pulsador rojo realiza la misma función que la tecla *Enter* o *Intro* en una consola de órdenes de un computador personal. Esta función es indicar al sistema que el usuario ha terminado de introducir el dato y que el programa puede recoger este dato y procesarlo. En la Imagen 4 se puede ver el diagrama de flujo que seguirá el programa que permite leer el dato del teclado y copiarlo al registro RA. En el primer paso de este diagrama se lee el valor del puerto 0x02. Si es cero significa que el usuario no ha validado todavía el dato, por lo que se vuelve a consultar el puerto 0x02. Mientras se ejecuta este bucle el usuario puede pulsar las teclas hexadecimales para introducir el dato deseado. Una vez el usuario pulsa el botón rojo, validando el dato introducido, el bucle finaliza y se lee el dato en el puerto 0x00. El hardware del teclado pone a 0 el contenido del puerto 0x02, dejando al sistema preparado para la lectura de un nuevo dato.

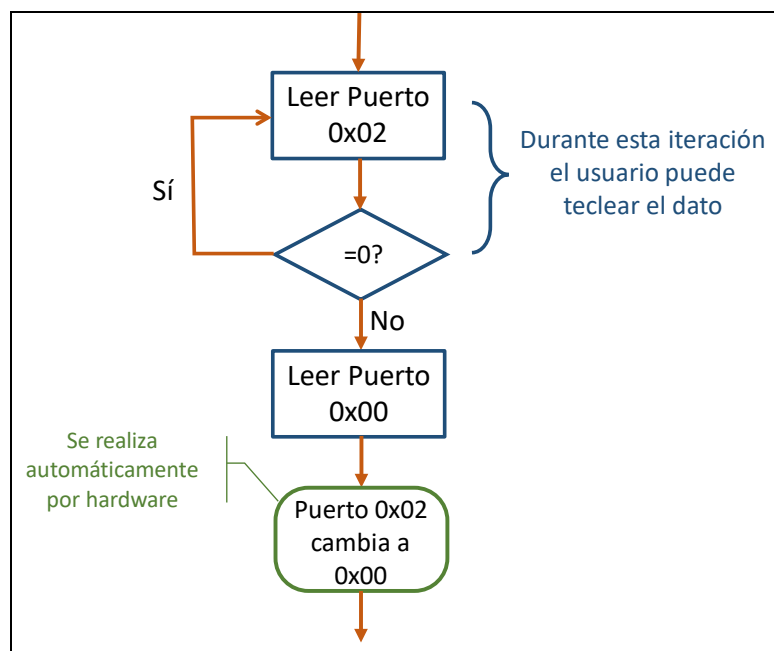
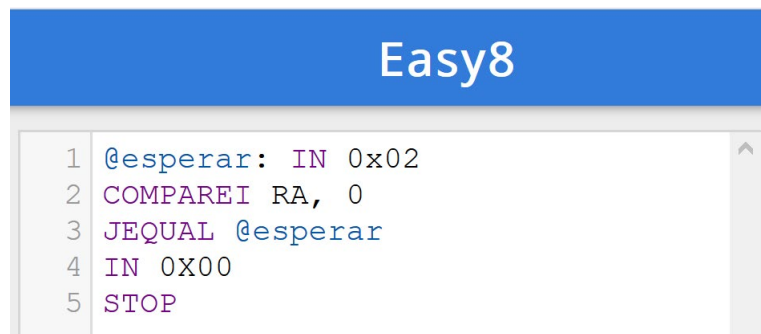


Imagen 4. Diagrama de flujo de lectura de un dato introducido a través del teclado hexadecimal.

En la Imagen 5 se muestra la secuencia de código ensamblador del Easy 8 para leer un dato de teclado. Es muy recomendable que accedas al simulador (easy8.webs.upv.es), introduzcas el programa en el editor, cambies a la vista simulador y ejecutes el programa, primero paso a paso, y después en continuo para verificar el funcionamiento del teclado hexadecimal y sus componentes asociados. Presta atención a los puertos de entrada/salida 0x00 y 0x02, y al registro RA.



```

Easy8
1 @esperar: IN 0x02
2 COMPAREI RA, 0
3 JEQUAL @esperar
4 IN 0X00
5 STOP

```

Imagen 5. Secuencia de código ensamblador para la lectura de un dato a través del teclado hexadecimal.

4.3 Ejercicios

Para asegurarnos que ha quedado claro y que eres capaz de leer un dato del teclado hexadecimal, te propongo dos ejercicios.

Ejercicio 1.

Responde a esta pregunta: Considerando que el valor almacenado en el puerto de E/S 0x00 es 0x00, ¿Cuál sería la evolución del display de eco si el usuario presiona la secuencia de teclas siguiente: 3, A, 5, 6, B, 7? Puedes usar el simulador para realizar este ejercicio.

Ejercicio 2.

Escribe un programa en lenguaje ensamblador del Easy 8 que lea un dato del teclado, lo muestre en el display de 2 dígitos (que está conectado al puerto 0x01), luego le sume 1 y muestre el resultado de nuevo en el display. Finalmente, el programa debe repetirse de forma infinita.

4.3.1 Soluciones a los ejercicios

Aquí tienes las soluciones a los ejercicios. Pero te pido por favor que antes de mirarlas, intentes resolverlos por ti mismo.

Solución al ejercicio 1:

La secuencia sería: 0x00 – 0x03 – 0x3A – 0xA5 – 0x56 – 0x6B – 0xB7



Solución al ejercicio 2:

```
1 @esperar: IN 0x02
2 COMPAREI RA, 0x00
3 JEQVAL @esperar
4 IN 0x00
5 OUT 0x01
6 INC RA
7 OUT 0x01
8 JUMF @esperar
9 STOP
```

5 Conclusiones

Ahora eres capaz de usar el teclado hexadecimal para leer un dato introducido por el usuario. Te recuerdo que el teclado del Easy 8 se compone de tres elementos: teclado hexadecimal, display de eco y pulsador rojo.

La forma de leer el dato del teclado es esperar en un bucle a que el usuario pulse el botón rojo. Esta espera se lleva a cabo leyendo el contenido del puerto de entrada/salida 0x02 hasta que tome valor distinto de 0. Finalizada esta espera, se copia el dato introducido por el usuario al registro RA del procesador leyendo el puerto de entrada/salida 0x00 (IN 0x00), que además pondrá a 0 el contenido del puerto de entrada/salida 0x02.

La técnica que hemos utilizado se llama *polling* o encuesta, ya que se trata de preguntarle a un periférico de forma repetida si tiene un dato disponible o si está preparado para recibir un dato. Esta es una de las posibles técnicas de sincronización entre el procesador y los periféricos.

6 Bibliografía

6.1 Libros:

Angulo Usategui, José María; “introducción a los computadores”. Madrid : Paraninfo, D.L. 1994.

Ors Carot, Rafael; Martí Campoy, Antonio; Gracia Morán, Joaquín; Pérez Jiménez, Alberto José; Saiz Adalid, Luis;: “Apuntes de informática industrial”. Editorial UPV, 2012

6.2 Referencias de fuentes electrónicas:

González Téllez, Alberto. (2009). Introducción al Easy 8. Disponible en: <http://hdl.handle.net/10251/5219>

González Carrizo, R. (2019). Simulación y almacenamiento de programas del Easy8 en Web. Disponible en: <http://hdl.handle.net/10251/124990>