

# Estructura de Computadores I

Grado en Ingeniería Informática

Soluciones para el emulador de la máquina elemental

## Introducción

El objetivo de este documento es presentar una batería de pruebas que permitirá verificar el correcto funcionamiento de la máquina elemental implementada durante las últimas sesiones de prácticas de la asignatura. Es importante realizar esta verificación, ya que os permitirá conocer si todas las fases de vuestro emulador (*fetch*, decodificación y ejecución) funcionan correctamente, detectando posibles errores en la implementación, que deberéis tener en cuenta de cara a la práctica final.

## Metodología

La batería de pruebas consta de tres tests. A continuación se explica cada uno de ellos, con sus correspondientes soluciones.

### Test 1

El primer test se utiliza para validar el correcto funcionamiento de la subrutina de decodificación, así como que está implementada como subrutina de librería. Para realizar este test, debéis abrir el fichero *test1.X68* y pegar el código correspondiente a vuestra subrutina en el lugar que se indica. Tras la ejecución, la solución se muestra en el siguiente cuadro:

CODIFICACIÓN EN HEXADECIMAL	EINSTRUCCIÓN	NUMCO(A3) EN HEXADECIMAL	D5 EN DECIMAL
043D	STORE	03	0
0155	JMZ	01	2
0C00	HALT	0A	4
00AA	JMI	00	6
0818	R1 -> R0	08	8
02FF	JMN	02	10
0804	CMP	05	12
06E1	LOAD	04	14
0820	R0 -> R1	09	16
0814	SUB	07	18
0810	ADD	06	20

El programa lee una secuencia de instrucciones de entrada y, tras esto, llama a la subrutina de decodificación. El resultado que devuelve vuestra subrutina se almacena en un vector de *words* llamado NUMCO, cuyo contenido en hexadecimal tras la ejecución es el que se muestra en la tabla.

## Test 2

El segundo test se utiliza para verificar que el programa deja los registros y la memoria correctamente tras la ejecución de cada una de las instrucciones. Para realizar este test, debéis sustituir la cabecera de vuestro emulador por la que se encuentra en el fichero *test2.X68*. Tras esto, es recomendable que utilicéis un *breakpoint* en la primera instrucción de la fase de *fetch* para detener la ejecución del programa tras emular cada instrucción, aunque también podéis ejecutar paso a paso todo el programa emulado. El contenido de los registros al final de la ejecución de cada instrucción debe ser el siguiente:

EDIR (Dec)	EINST	EIR (Hex)	EPC (Hex)	ER0 (Hex)	ER1 (Hex)	ESR (Hex)	M(10 Dec) (Hex)	M(11 Dec) (Hex)	M(12 Dec) (Hex)
0	LOAD 10	060A	0001	0000	0001	0000	0001	0002	0000
1	R1 -> R0	0818	0002	0001					
2	LOAD 11	060B	0003		0002				
3	SUB	0814	0004	0FFF		0002			
4	JMN 7	0207	0007						
7	R0 -> R1	0820	0008		0FFF				
8	STORE 12	040C	0009						0FFF
9	HALT	0C00	000A						

Los registros están definidos a partir de la posición *111A* hexadecimal y la memoria a partir de la *1114* hexadecimal.

## Test 3

El último test emula la ejecución de una secuencia larga de instrucciones. El objetivo es comprobar que los registros contienen, tras la ejecución de toda la secuencia, la siguiente información:

EPC (Hex)	ER0 (Hex)	ER1 (Hex)	ESR (Hex)	M(21 Dec) (Hex)	M(22 Dec) (Hex)	M(23 Dec) (Hex)	M(24 Dec) (Hex)
0015	08A6	00F2	0002	00FA	07B4	00F2	0001

Para ejecutar este test, debéis seguir el mismo procedimiento que en la prueba anterior: sustituid la cabecera de vuestro emulador por la que se encuentra en el fichero *test3.X68*. Tras la ejecución, la memoria debe contener lo indicado en la tabla definida más arriba. Los registros están definidos a partir de la posición *1132* hexadecimal, mientras que la memoria está definida a partir de la *112A* hexadecimal.