

ET10a (Unitat de Control General)

Exercicis per avaluar objectius de nivell B

Objectius: 10.1 i 10.4

Exercici 10.1. (Objectiu 10.1)

Indiqueu quines de les següents afirmacions són certes (pot haver més d'una):

- a) Gràcies al seqüenciament implícit qualsevol instrucció pot trencar el seqüenciament del programa a banda de fer la seva funció pròpia.
- b) Gràcies al seqüenciament implícit podem reduir el nombre de bits necessaris per expressar una instrucció.
- c) Gràcies al seqüenciament implícit podem saltar a qualsevol instrucció del programa
- d) El seqüenciament implícit assumeix que, donat un llistat d'instruccions, després d'una instrucció, com a norma general sempre s'executarà la següent instrucció.
- e) En un sistema amb seqüenciament implícit no són necessàries les instruccions de salt o ruptura de seqüència.

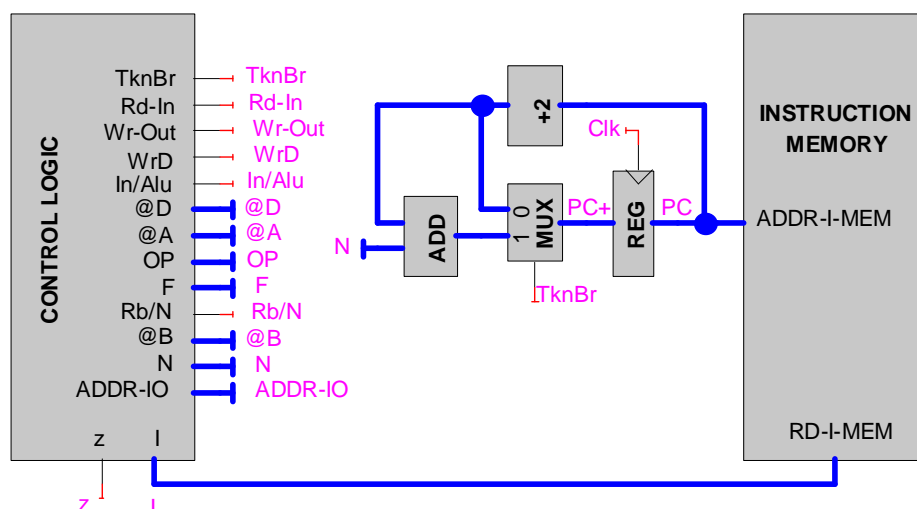
Exercici 10.2. (Objectiu 10.1)

Indiqueu quines de les següents afirmacions són certes (pot haver més d'una):

- a) Gràcies als salts relatius al valor que emmagatzemem al registre comptador de programa, PC, podem saltar des de qualsevol instrucció d'un programa a qualsevol altra.
- b) En un sistema on els salts són relatius al valor que emmagatzemem al registre comptador de programa, PC, el nombre de posicions a la seqüència d'instruccions que es pot saltar depèn del rang de valors que es reservin a la instrucció per expressar el desplaçament respecte al valor del PC.
- c) En un sistema on els salts són relatius al valor que emmagatzemem al registre comptador de programa, PC, no es pot saltar a instruccions anteriors, només a instruccions que es troben després de l'actual.
- d) En un sistema on els salts són relatius al valor que emmagatzemem al registre comptador de programa, PC, no calen instruccions especials de salt o ruptura de seqüència.

Exercici 10.3. (Objectiu 10.4)

A continuació trobareu el circuit de la Unitat de Control General (UCG):



Tenint en compte que treballem amb un llenguatge màquina amb instruccions de 16 bits que s'emmagatzemen a la memòria d'instruccions i que aquest llenguatge només té instruccions de salt condicionals, que produeixen salts relatius al valor actual del PC, digues, per cada una de les situacions que es presenten a continuació, quina de les següents afirmacions és certa (pot haver més d'una):

- a) *Situació:* el valor del senyal PC al cicle i és $PC[i]=0x0A38$ i al cicle $i+4$ és $PC[i+4]=0x0A3A$ i sabem que entre el cicle i i el cicle $i+3$ només s'ha executat una vegada una instrucció de salt:
- Durant el cicle en el que s'ha executat la instrucció de salt, el valor del senyal *TknBr* s'ha estabilitzat a 0 i per tant no s'ha produït cap ruptura al seqüenciamnt del programa.
 - La instrucció que es troba a la posició de la memòria d'instruccions $0x0A38$ no pot ser una instrucció de salt.
 - Durant el cicle en el que s'ha executat la instrucció de salt, el valor del senyal *TknBr* s'ha estabilitzat a 1 i N ha pres per valor un nombre enter positiu codificat en Ca2.
 - Durant el cicle en el que s'ha executat la instrucció de salt, el valor del senyal *TknBr* s'ha estabilitzat a 1 i N ha pres per valor un nombre enter negatiu codificat en Ca2.
 - S'ha realitzat un salt absolut i N ha pres per valor $0x0A3A$.
- b) *Situació:* el valor del senyal PC al cicle i és $PC[i]=0x5F2A$ i al cicle $i+6$ és $PC[i+6]=0x5F30$ i sabem que entre el cicle i i el cicle $i+6$ només s'ha executat una instrucció de salt (amb un desplaçament diferent a 1, $N \neq 1$):
- Durant el cicle en el que s'ha executat la instrucció de salt, el valor del senyal *TknBr* s'ha estabilitzat a 0 i per tant no s'ha produït cap ruptura al seqüenciamnt del programa.
 - La instrucció que es troba a la posició de la memòria d'instruccions $0x0A38$ no pot ser una instrucció de salt.
 - Durant el cicle en el que s'ha executat la instrucció de salt, el valor del senyal *TknBr* s'ha estabilitzat a 1 i N ha pres per valor un nombre enter positiu codificat en Ca2.
 - Durant el cicle en el que s'ha executat la instrucció de salt, el valor del senyal *TknBr* s'ha estabilitzat a 1 i N ha pres per valor un nombre enter negatiu codificat en Ca2.

ET10b (Unitat de Control General)**Exercicis per avaluar objectius de nivell B****Objectius:** 10.5, 10.6 i 10.7**Exercici 10.4.** (Objectiu 10.5)

Assembleu les següents instruccions del llenguatge màquina SISA. Indiqueu el resultat en hexadecimal com l'exemple. Si hi ha alguna instrucció incorrecta (per algun motiu no es pot assembleu), indiqueu-ho posant **NA**.

Exemple: **OR R6, R3, R5** Solució: **0x0CCD**

- | | | |
|---------------------|-------------------|--------------------|
| a) AND R6, R4, R7 | e) MOVHI R2, -128 | i) MOVI R3, 0xF3 |
| b) CMPLT R7, R2, R3 | f) BNZ R4, -30 | j) ADDI R0, R1, 32 |
| c) BZ R4, 14 | g) NOT R0, R5 | k) SHL R4, R2, R5 |
| d) IN R5, 140 | h) OUT R0, 27 | |

Exercici 10.5. (Objectiu 10.5)

Desassembleu les següents instruccions del llenguatge màquina SISA. Indiqueu el resultat amb els mnemotècnics corresponents. Indiqueu posant **NA** si hi ha alguna instrucció incorrecta (el codi binari no correspon a cap instrucció del llenguatge màquina SISA).

- | | | |
|-----------|-----------|-----------|
| a) 0x0459 | c) 0x96AB | e) 0x89D9 |
| b) 0x247D | d) 0x11F5 | f) 0x0772 |

Exercici 10.6. (Objectiu 10.6)

Indiqueu quin és l'estat del computador (el contingut dels registres del banc de registres i les posicions de memòria modificades, dels ports de sortida modificats, el registre PC, etc) després de l'execució de cadascuna de les següents instruccions. Supposeu que el contingut inicial dels registres i de les posicions de memòria és igual per a totes les instruccions i és el següent:

Registres		Ports d'entrada		PC	0x1EB8
Rx	Contingut	Port	Contingut		
R0	0x366A	Input[0x00]	0XAF03		
R1	0x8A0C	Input[0x01]	0xC889		
R2	0x6551	Input[0x02]	0xDADA		
R3	0x0000	Input[0x03]	0x730D		
R4	0xFFFF	Input[0x04]	0x1234		
R5	0x0001	Input[0x05]	0x0001		
R6	0x0002		
R7	0xFFFE				

Utilitzeu la següent nomenclatura per indicar els canvis (serà la mateixa que utilitzareu a Atenea):

- 1) Modificació d'un registre: **Rn=0xvalor**
- 2) Modificació de dades de sortida: **Output[0xport]=0xvalor**
- 3) Modificació del PC: **PC=0xvalor**

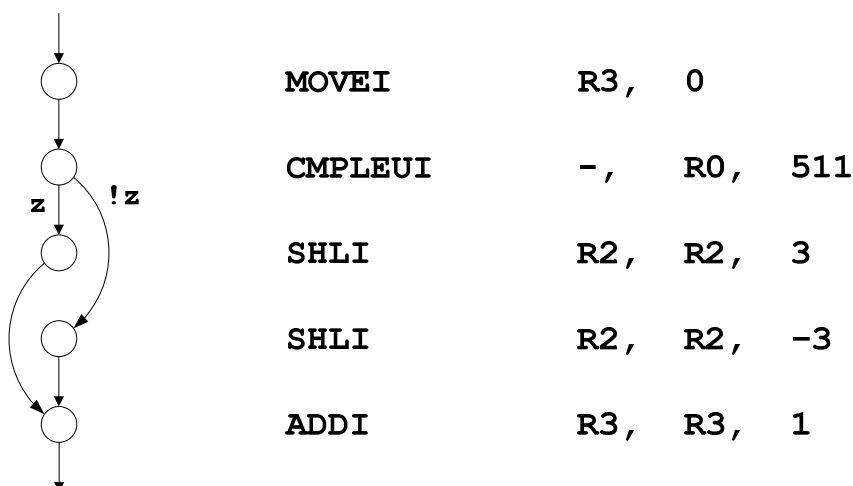
Utilitzeu 4 dígits hexadecimal per indicar *valor* i 2 dígits per a *port*. Si una instrucció modifica més d'una dada, indiqueu-ho.

Exemple: **OR R6, R3, R5** Solució: **R6=0x0001 PC=0x1EB9**

- | | | |
|-------------------|--------------------|---------------------|
| a) SHA R5, R1, R7 | d) ADDI R2, R3, 10 | g) IN R6, 2 |
| b) SHL R4, R4, R7 | e) MOVHI R2, 0x3D | h) BNZ R4, 13 |
| c) SUB R2, R1, R2 | f) OUT 0x2A, R7 | i) CMPLT R1, R4, R2 |

Exercici 10.7. (Objectiu 10.7)

Suposeu el següent graf de Moore corresponent a una Unitat de Control de propòsit específic (UCE) dissenyada per controlar la Unitat de Propòsit General (UPG) presentada a l'assignatura:



Substituïu la UCE per la Unitat de Control General (UCG) presentada a l'assignatura. Per fer-ho només cal indicar les instruccions en llenguatge ensamblador SISA que falten, per tal que la nova UCG tingui la mateixa funcionalitat que la UCE original.

Llenguatge SISA
MOVI R3, 0
MOVI R7, 0xFF
CMPLT R7, R0, R7
MOVI R6, 3
____ R7, ____
MOVI R6, -3
ADDI R3, R3, 1

Exercici 10.8. (Objectiu 10.7)

En els següents apartats trobareu seqüències d'instruccions escrites en pseudocodi en llenguatge d'alt nivell. Ompliu els forats que falten amb les instruccions adequades en llenguatge ensamblador SISA corresponents a les funcions que realitzen les sentències en alt nivell. Suposeu que tots els valors dels registres i les constants són nombres naturals.

a)

```
R2 = 9;
if (R3<=7) {
    R2 = R2 + 1;
}
R7 = R2 + R3;
```

Llenguatge SISA-I	
MOVI	R2, 9
MOVI	R0, 7
CMPLEU	R1, R3, R0
ADD	R7, R2, R3

b)

```
R4 = -2;
while (R3>=R2) {
    R2 = R2 - 1;
    R3 = R3 >> 2;
}
R7 = R2 + R3;
```

Llenguatge SISA-I	
MOVI	R4, -2
BNZ	R1, 4
ADDI	R2, R2, -1
BZ	R1, -4
ADD	R7, R2, R3

*>> és l'operació de desplaçament (shift) cap la dreta.

c)

```
R2 = 9;
if ((R2>R1) && (R3<=R4)) {
    R1 = R1 * 2;
} else {
    R4 = R4 - 1;
}
R7 = R2 + R3;
```

Llenguatge SISA-I	
MOVI	R2, 9
BNZ	R0, 5
CMPLEU	R0, R3, R4
ADD	R1, R1, R1
ADDI	R4, R4, -1
ADD	R7, R2, R3

*&& és l'operació AND lògica.

ET10b (Unitat de Control General)

Exercicis per avaluar objectius de nivell A

Objectius: 10.8

(Recordeu que l'objectiu amb l'asterisc cal fer-lo a casa i portar-lo resolt a classe)

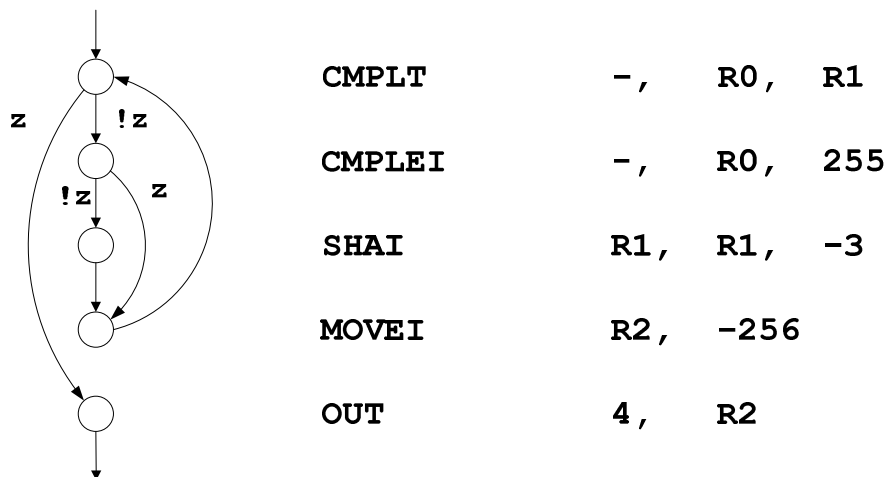
Exercici 10.9. (Objectiu 10.8)

Proposeu una instrucció o un conjunt d'instruccions del llenguatge ensamblador SISA per a realitzar les següents accions expressades textualment.

- Donat el valor emmagatzemat al registre R3, que s'interpreta com un nombre natural codificat en binari, dividiu-lo entre 8 i emmagatzemeu el resultat al registre R6.
- Donat el valor d'un nombre enter emmagatzemat al registre R7, codificat en Ca2, dividiu-lo entre 4 i emmagatzemeu el resultat al registre R0.
- Donat el valor d'un nombre enter emmagatzemat al registre R0, codificat en Ca2, multipliqueu-lo per 16 i deixeu el resultat al mateix registre.
- Donat un valor emmagatzemat al registre R4, volem emmagatzemar a R5 un valor tal que tots els seus bits valguin 0, excepte el bit de la posició 7 (recorda que el bit de menys pes es considera que està a la posició 0) que ha de valer el mateix que el bit de la posició 7 del registre R4. És a dir: $R5\langle i \rangle = 0$, si $i \neq 7$, i $R5\langle 7 \rangle = R4\langle 7 \rangle$.
- Donat un valor al port 5 d'entrada, sumar-li 40 i retornar el resultat a través del port 60 de sortida.
- Emmagatzemeu el valor 514 al registre R4.

(*) Exercici 10.10. (Objectiu 10.8)

Suposeu el següent graf de Moore corresponent a una Unitat de Control de propòsit específic (UCE) dissenyada per controlar la Unitat de Propòsit General (UPG) presentada a l'assignatura:



A continuació substituïm la UCE per la Unitat de Control General (UCG) presentada a l'assignatura. Indiqueu un conjunt d'instruccions en llenguatge ensamblador SISA que garanteixi que la nova UCG té la mateixa funcionalitat que la UCE original.

Exercici 10.11. (Objectiu 10.8)

A continuació trobareu una seqüència d'instruccions escrita en pseudocodi en un llenguatge d'alt nivell. Escriviu un conjunt d'instruccions en llenguatge ensamblador SISA que realitzi la mateixa funció que es descriu mitjançant les sentències en alt nivell. Suposeu que tots els valors del registres i les constants són nombres enters.

```

R7 = INPUT[20];
R2 = -3;
if ((R2>R7) || (R3<=R4)) {
    R2 = R2 + 1;
    while (R3>R2) {
        R2 = R2 >> 2;
        R3 = R3 - 1;
    }
} else {
    R4 = R4 - 1;
}
OUTPUT[3] = R2 + R3;

```

* >> és l'operació de desplaçament (shift) cap la dreta i || és l'operació OR lògica.

Solucions ET10 (Unitat de Control General)

ET10a

Exercici 10.1.

Les afirmacions correctes són la b) i la d). Les afirmacions a), c) i e) són falses.

Exercici 10.2.

La única afirmació correcta és la b).

Exercici 10.3.

a) Les respostes correctes són la ii) i la iv).

b) La única resposta correcta és la i).

ET10b

Exercici 10.4.

a) AND R6, R4, R7	-	0x09F0
b) CMPLT R7, R2, R3	-	0x14F8
c) BZ R4, 14	-	0x880E
d) IN R5, 140	-	0xAA8C
e) MOVHI R2, -128	-	0x9580
f) BNZ R4, -30	-	0x89E2
g) NOT R0, R5	-	0x0A03
h) OUT R0, 27	-	NA
i) MOVI R3, 0xF3	-	0x96F3
j) ADDI R0, R1, 32	-	NA
k) SHL R4, R2, R5	-	0x0567

Exercici 10.5.

a) 0x0459	-	OR R3, R2, R1
b) 0x247D	-	ADDI R1, R2, -3
c) 0x96AB	-	MOVI R3, -85
d) 0x11F5	-	CMPLEU R6, R0, R7
e) 0x89D9	-	BNZ R4, -39
f) 0x0772	-	XOR R6, R3, R5

Exercici 10.6.

- a) **R5=0xE283** **PC=0x1EBA**
- b) **R4=0x3FFF** **PC=0x1EBA**
- c) **R2=0x24BB** **PC=0x1EBA**
- d) **R2=0x000A** **PC=0x1EBA**
- e) **R2=0x3D51** **PC=0x1EBA**
- f) **Output[0x2A]=0xFFFE** **PC=0x1EBA**
- g) **R6=0xDADA** **PC=0x1EBA**
- h) **PC=0x1ED4**
- i) **R1=0x0001** **PC=0x1EBA**

Exercici 10.7.

Llenguatge SISA-I
MOVI R3, 0
MOVI R7, 0xFF
MOVHI R7, 0x01
CMPLEU R7, R0, R7
BNZ R7, 3
MOVI R6, 3
SHL R2, R2, R6
BZ R7, 2
MOVI R6, -3
SHL R2, R2, R6
ADDI R3, R3, 1

Exercici 10.8.

a)

Llenguatge SISA-I
MOVI R2, 9
MOVI R0, 7
CMPLEU R1, R3, R0
BZ R1, 1
ADDI R2, R2, 1
ADD R7, R2, R3

b)

Llenguatge SISA-I		
MOVI	R4,	-2
CMPLTU	R1, R3,	R2
BNZ	R1,	3
ADDI	R2, R2,	-1
SHL	R3, R3,	R4
BZ	R1,	-5
ADD	R7, R2,	R3

c)

Llenguatge SISA-I		
MOVI	R2,	9
CMPLEU	R0, R2,	R1
BNZ	R0,	4
CMPLEU	R0, R3,	R4
BZ	R0,	2
ADD	R1, R1,	R1
BNZ	R0,	1
ADDI	R4, R4,	-1
ADD	R7, R2,	R3