

Secondo Esonero di “Reti Logiche e Calcolatori” del 13/5/2016 – Traccia A

ESERCIZIO: Estendere il set di istruzioni della macchina ad accumulatore con l'operazione **VMOD X**, definita come segue.

A partire dalla locazione X+1 della RAM è memorizzato un vettore di L elementi, dove L è il valore contenuto in M[X].

L'istruzione modificherà il vettore sostituendo ad ogni elemento negativo V[i] di V il valore $-(2 \cdot V[i])$ e memorizzerà il numero di sostituzioni effettuate nell'accumulatore.

La figura sulla destra mostra un esempio dello stato della memoria e dei registri prima e dopo l'esecuzione dell'istruzione.

PRIMA				DOPO			
X	:			X	:		
1052	L	1052	8	1052	L	1052	8
	V[0]	1053	3		V[0]	1053	3
	V[1]	1054	-2		V[1]	1054	4
	V[2]	1055	-11	AC	V[2]	1055	22
	V[3]	1056	3	3	V[3]	1056	3
	V[4]	1057	8		V[4]	1057	8
	V[5]	1058	-6		V[5]	1058	12
	V[6]	1059	9		V[6]	1059	9
	V[7]	1060	12		V[7]	1060	12
	:				:		

CODICE RTL

```

μ1    IRX → MAR;
μ2    M[MAR] → MBR, INCR(MAR) → MAR;
μ3    MBR → T1, 0 → T2;
1: if OR(T1) = 1 then
μ4        M[MAR] → MBR;
μ5        MBR → B;
        if B31 = 1 then
μ6            -B → MBR, INCR(T2) → T2;
μ7            SHL(MBR) → MBR;
μ8            MBR → M[MAR], INCR(MAR) → MAR, DECR(T1) → T1, goto 1;
        else
μ9            INCR(MAR) → MAR, DECR(T1) → T1, goto 1;
        fi
    else
μ10       T2 → AC;
        fi

```

Modifiche architetturali

- aggiunta funzione di incremento al registro MAR
- aggiunta funzione di scorrimento sinistro al registro MBR
- aggiunta funzione di azzeramento al registro T2
- aggiunta del segnale beta B₃₁ collegato al bit più significativo del registro B

Tabella di ROM della parte di controllo

COP	OR(T1)	B ₃₁	y ₂	y ₁	y ₀	y' ₂	y' ₁	y' ₀	μ
00001001	-	-	0	0	0	0	0	1	μ ₁
00001001	-	-	0	0	1	0	1	0	μ ₂
00001001	-	-	0	1	0	0	1	1	μ ₃
00001001	1	-	0	1	1	1	0	0	μ ₄
00001001	1	-	1	0	0	1	0	1	μ ₅
00001001	1	1	1	0	1	1	1	0	μ ₆
00001001	1	1	1	1	0	1	1	1	μ ₇
00001001	1	1	1	1	1	0	1	1	μ ₈
00001001	1	0	1	0	1	0	1	1	μ ₉
00001001	0	-	0	1	1	0	0	0	μ ₁₀ , Z _{ir} = 1

Secondo Esonero di “Reti Logiche e Calcolatori” del 13/5/2016 – Traccia B

ESERCIZIO: Estendere il set di istruzioni della macchina ad accumulatore con l'operazione **VMOD X**, definita come segue.

A partire dalla locazione X+1 della RAM è memorizzato un vettore di L elementi, dove L è il valore contenuto in M[X].

L'istruzione modificherà il vettore sostituendo ad ogni elemento dispari V[i] di V il valore (V[i]*2)+3 e memorizzerà il numero di sostituzioni effettuate nell'accumulatore.

La figura sulla destra mostra un esempio dello stato della memoria e dei registri prima e dopo l'esecuzione dell'istruzione.

PRIMA				DOPO			
X	L			X	L		
1052				1052			
	V[0]	1053	8		V[0]	1053	12
	V[1]	1054	12		V[1]	1054	9
	V[2]	1055	3	AC	V[2]	1055	14
	V[3]	1056	14	4	V[3]	1056	4
	V[4]	1057	4		V[4]	1057	17
	V[5]	1058	7		V[5]	1058	13
	V[6]	1059	5		V[6]	1059	10
	V[7]	1060	10		V[7]	1060	9
			3				

CODICE RTL

```

μ1    IRx → MAR;
μ2    M[MAR] → MBR, INCR(MAR) → MAR;
μ3    MBR → T1, 0 → T2, 3 → A;
1: if OR(T1) = 1 then
μ4        M[MAR] → MBR;
μ5        MBR → AC;
        if AC0 = 1 then
μ6            MBR → B, INCR(T2) → T2;
μ7            SHL(B) → B;
μ8            A + B → MBR;
μ9            MBR → M[MAR], INCR(MAR) → MAR, DECR(T1) → T1, goto 1;
        else
μ10        INCR(MAR) → MAR, DECR(T1) → T1, goto 1;
        fi
    else
μ11        T2 → AC;
    fi

```

Modifiche architetturali

- aggiunta funzione di incremento al registro MAR
- aggiunta funzione di scorrimento sinistro al registro B
- aggiunta funzione di caricamento della costante 3 al registro A
- aggiunta funzione di azzeramento al registro T2
- aggiunta del segnale beta AC₀ collegato al bit meno significativo del registro AC

Tabella di ROM della parte di controllo

COP	OR(T1)	AC ₀	y ₃	y ₂	y ₁	y ₀	y' ₃	y' ₂	y' ₁	y' ₀	μ
00001001	–	–	0	0	0	0	0	0	0	1	μ ₁
00001001	–	–	0	0	0	1	0	0	1	0	μ ₂
00001001	–	–	0	0	1	0	0	0	1	1	μ ₃
00001001	1	–	0	0	1	1	0	1	0	0	μ ₄
00001001	1	–	0	1	0	0	0	1	0	1	μ ₅
00001001	1	1	0	1	0	1	0	1	1	0	μ ₆
00001001	1	1	0	1	1	0	0	1	1	1	μ ₇
00001001	1	1	0	1	1	1	1	0	0	0	μ ₈
00001001	1	1	1	0	0	0	0	0	1	1	μ ₉
00001001	1	0	0	1	0	1	0	0	1	1	μ ₁₀
00001001	0	–	0	0	1	1	0	0	0	0	μ ₁₁ , Z _{ir} = 1

Secondo Esonero di “Reti Logiche e Calcolatori” del 13/5/2016 – Traccia C

ESERCIZIO: Estendere il set di istruzioni della macchina ad accumulatore con l'operazione **VMOD X**, definita come segue.

A partire dalla locazione X+1 della RAM è memorizzato un vettore di L elementi, dove L è il valore contenuto in M[X].

L'istruzione modificherà il vettore sostituendo ad ogni elemento positivo V[i] di V il valore $(V[i]+16)/2$ e memorizzerà il numero di sostituzioni effettuate nell'accumulatore.

La figura sulla destra mostra un esempio dello stato della memoria e dei registri prima e dopo l'esecuzione dell'istruzione.

PRIMA			DOPO		
X	:		X	:	
1052	L	1052	1052	L	1052
	V[0]	1053		V[0]	1053
	V[1]	1054		V[1]	1054
	V[2]	1055	AC	V[2]	1055
	V[3]	1056	3	V[3]	1056
	V[4]	1057		V[4]	1057
	V[5]	1058		V[5]	1058
	V[6]	1059		V[6]	1059
	V[7]	1060		V[7]	1060
	:			:	

CODICE RTL

```

μ1    IRx → MAR;
μ2    M[MAR] → MBR, INCR(MAR) → MAR;
μ3    MBR → T1, 0 → T2, 16 → A;
1: if OR(T1) = 1 then
μ4        M[MAR] → MBR;
μ5        MBR → B;
        if B31 = 0 then
μ6            A + B → MBR, INCR(T2) → T2;
μ7            SHR(MBR) → MBR;
μ8            MBR → M[MAR], INCR(MAR) → MAR, DECR(T1) → T1, goto 1;
        else
μ9            INCR(MAR) → MAR, DECR(T1) → T1, goto 1;
        fi
    else
μ10       T2 → AC;
    fi

```

Modifiche architetturali

- aggiunta funzione di incremento al registro MAR
- aggiunta funzione di scorrimento destro al registro MBR
- aggiunta funzione di caricamento della costante 16 al registro A
- aggiunta funzione di azzeramento al registro T2
- aggiunta del segnale beta B₃₁ collegato al bit più significativo del registro B

Tabella di ROM della parte di controllo

COP	OR(T1)	B ₃₁	y ₂	y ₁	y ₀	y' ₂	y' ₁	y' ₀	μ
00001001	–	–	0	0	0	0	0	1	μ ₁
00001001	–	–	0	0	1	0	1	0	μ ₂
00001001	–	–	0	1	0	0	1	1	μ ₃
00001001	1	–	0	1	1	1	0	0	μ ₄
00001001	1	–	1	0	0	1	0	1	μ ₅
00001001	1	1	1	0	1	1	1	0	μ ₆
00001001	1	1	1	1	0	1	1	1	μ ₇
00001001	1	1	1	1	1	0	1	1	μ ₈
00001001	1	0	1	0	1	0	1	1	μ ₉
00001001	0	–	0	1	1	0	0	0	μ ₁₀ , Z _{ir} = 1

Secondo Esonero di “Reti Logiche e Calcolatori” del 13/5/2016 – Traccia D

ESERCIZIO: Estendere il set di istruzioni della macchina ad accumulatore con l'operazione **VMOD X**, definita come segue.

A partire dalla locazione X+1 della RAM è memorizzato un vettore di L elementi, dove L è il valore contenuto in M[X].

L'istruzione modificherà il vettore sostituendo ad ogni elemento pari V[i] di V il valore (V[i]/2)-4 e memorizzerà il numero di sostituzioni effettuate nell'accumulatore.

La figura sulla destra mostra un esempio dello stato della memoria e dei registri prima e dopo l'esecuzione dell'istruzione.

PRIMA				DOPO			
X	L			X	L		
1052				1052			
	V[0]	1053	8		V[0]	1053	8
	V[1]	1054	13		V[1]	1054	13
	V[2]	1055	12		V[2]	1054	2
	V[3]	1055	14		V[2]	1055	3
	V[4]	1056	3		V[3]	1056	3
	V[5]	1057	9		V[4]	1057	9
	V[6]	1058	7		V[5]	1058	7
	V[7]	1059	9		V[6]	1059	9
		1060	12		V[7]	1060	2

CODICE RTL

```

μ1      IRx → MAR;
μ2      M[MAR] → MBR, INCR(MAR) → MAR;
μ3      MBR → T1, 0 → T2, 4 → B;
1: if OR(T1) = 1 then
μ4          M[MAR] → MBR;
μ5          MBR → AC;
           if AC0 = 0 then
μ6              MBR → A, INCR(T2) → T2;
μ7              SHR(A) → A;
μ8              A - B → MBR;
μ9              MBR → M[MAR], INCR(MAR) → MAR, DECR(T1) → T1, goto 1;
           else
μ10          INCR(MAR) → MAR, DECR(T1) → T1, goto 1;
           fi
           else
μ11          T2 → AC;
           fi

```

Modifiche architetturali

- aggiunta funzione di incremento al registro MAR
- aggiunta funzione di scorrimento destro al registro A
- aggiunta funzione di caricamento della costante 4 al registro B
- aggiunta funzione di azzeramento al registro T2
- aggiunta del segnale beta AC₀ collegato al bit meno significativo del registro AC

Tabella di ROM della parte di controllo

COP	OR(T1)	AC ₀	y ₃	y ₂	y ₁	y ₀	y' ₃	y' ₂	y' ₁	y' ₀	μ
00001001	-	-	0	0	0	0	0	0	0	1	μ ₁
00001001	-	-	0	0	0	1	0	0	1	0	μ ₂
00001001	-	-	0	0	1	0	0	0	1	1	μ ₃
00001001	1	-	0	0	1	1	0	1	0	0	μ ₄
00001001	1	-	0	1	0	0	0	1	0	1	μ ₅
00001001	1	1	0	1	0	1	0	1	1	0	μ ₆
00001001	1	1	0	1	1	0	0	1	1	1	μ ₇
00001001	1	1	0	1	1	1	1	0	0	0	μ ₈
00001001	1	1	1	0	0	0	0	0	1	1	μ ₉
00001001	1	0	0	1	0	1	0	0	1	1	μ ₁₀
00001001	0	-	0	0	1	1	0	0	0	0	μ ₁₁ , Z _{ir} = 1