Lezione 3 – Esempi di programmi in linguaggio macchina LC-2

Architettura degli elaboratori

Modulo 2 – Linguaggio macchina

Unità didattica 2 - CPU LC-2

Nello Scarabottolo

Università degli Studi di Milano - Ssri - CDL ONLINE

Esempio sui modi di indirizzamento

```
; programma di uso dei modi di indirizzamento
```

; dove E=cifra esadecimale (corrisponde a 4 bit)

```
1 LEA R1, x30F4 ; carica 30F4 in R1
```

2 **ADD R2**, **R1**, **x000E** ; somma 14 a R1 (in R2 3102)

3 **ST R2**, **x30F4** ; scrive R2 in 30F4

4 **AND R2**, **R2**, **x0000** ; azzera R2

5 **ADD R2, R2, x0005** ; inizializza R2 a 5

6 **STR R2**, **R1**, **x000E** ; scrive R2 in M(R1+14=3102)

7 **LDI R3**, **x30F4** ; scrive M(M(30F4))=5 in R3

[;] situato in memoria a partire da x30F6

[;] usa notazione esadecimale: xEEEE

CPU e memoria all'inizio del programma R0 x0000 0 R4 x0000 U R1 x0000 0 R5 x0000 0 R2 x0000 0 R6 x0000 0 R2 x0000 0 R7 x0000 0 PC x30F6 12534 IR x0000 0 x30F4 000000000000000 x0000 - x30F5 <u>000000000000000</u> x0000 LEA ADD ST AND ADD STR R1, x30F4 R2, R1, x000E → x30F6 1110<mark>001</mark>011110100 xE2F4 0001<mark>010</mark>001<mark>1</mark>01110 x30F7 x146E x30F8 0011010011110100 x34F4 R2, x30F4 x30F9 0101010010100000 x54A0 R2, R2, x0000 R2, R2, x0005 R2, R1 x000E = x30FA 0001<mark>010</mark>010<mark>1</mark>00101 x14A5 x30FB 0111010001001110 x744E STR = x30FC 1010011011110100 xA6F4 LDI R3, x30F4 - x30FD 0000000000000000 x0000 - x30FE 000000000000000 x0000 x30FF 00000000000000000 x0000 x3100 000000000000000 x0000 - x3101 000000000000000 x0000 - x3102 000000000000000 x0000

		С	opo is	truzione 1
LEA R	1, x30F4 ;cari	ca 30F4 in R	1	
RO	x0000 0	R4 x0000 0	PC	x30F7 12535
Rl	x30F4 12532	R5 x0000 0	IR	xE2F4 -7436
R2	x0000 0	R6 x0000 0	CC	P
R3	x0000 0	R7 x0000 0		
= x30F4	00000000000000000	x0000		
= x30F5	00000000000000000	x0000		
= x30F6	1110001011110100	xE2F4	LEA	R1, x30F4
→ x30F7	0001010001101110	x146E	ADD	R2, R1, x000E
= x30F8	0011010011110100	x34F4	ST	R2, x30F4
= x30F9	0101010010100000	x54A0	AND	R2, R2, x0000
= x30FA	0001010010100101	x14A5	ADD	R2, R2, x0005
= x30FB	0111010001001110	x744E	STR	R2, R1 x000E
= x30FC	1010011011110100	xA6F4	LDI	R3, x30F4
= x30FD	000000000000000000	x0000		
= x30FE	00000000000000000	x0000		
= x30FF	00000000000000000	x0000		
= x3100	00000000000000000	x0000		
= x3101	00000000000000000	x0000		
= x3102	00000000000000000	x0000		

Dopo istruzione 2

```
ADD R2, R1, x000E ; somma 14 a R1 (in R2 3102)
     R0 x0000 0
                         R4 x0000 0
                                            PC x30F8 12536
                        R5 x0000 0
                                            IR x146E 5230
     R1 x30F4 12532
     R2 x3102 12546
                        R6 x0000 0
     R3 x0000 0
                        R7 x0000 0
  x30F4 0000000000000000 x0000
  x30F5 0000000000000000 x0000
  x30F6 1110001011110100 xE2F4
                                         LEA
                                                R1, x30F4
  = x30F7
          0001010001101110 x146E
                                         ADD
                                                R2, R1, x000E
  → x30F8 0011010011110100 x34F4
                                         ST
                                                R2, x30F4
  x30F9 0101010010100000 x54A0
                                         AND
                                                R2, R2, x0000
  x30FA 0001010010100101 x14A5
                                         ADD
                                                R2, R2, x0005
  x30FB
          0111010001001110 x744E
                                         STR
                                                R2, R1 x000E
  x30FC 1010011011110100 xA6F4
                                         LDI
                                                R3, x30F4
          000000000000000 x0000
  x30FD
  x30FE 0000000000000000 x0000
  x30FF
          000000000000000 x0000
          000000000000000 x0000
  x3100
  x3101 000000000000000 x0000
  x3102 000000000000000 x0000
```

Dopo istruzione 3

```
ST
       R2, x30F4; scrive R2 in 30F4
                         R4 x0000 0
      R0 x0000 0
                                            PC x30F9 12537
      R1 x30F4 12532
                         R5 x0000
                                             IR x34F4 13556
                                   0
                                             CC P
      R2 x3102 12546
                         R6 x0000 0
                         R7 x0000 0
      R3 x0000 0
  x30F4 0011000100000010 x3102
  x30F5 0000000000000000 x0000
  x30F6 1110001011110100 xE2F4
                                          LEA
                                                R1, x30F4
  = x30F7
          0001010001101110 x146E
                                          ADD
                                                R2, R1, x000E
  = x30F8
          0011010011110100
                                          ST
                                                 R2, x30F4
                                                R2, R2, x0000
  x30F9 0101010010100000 x54A0
                                          AND
  x30FA 0001010010100101 x14A5
                                                R2, R2, x0005
                                                R2, R1 x000E
  x30FB 0111010001001110 x744E
                                          STR
  x30FC
          1010011011110100
                                          LDI
                                                R3, x30F4
                          xA6F4
          0000000000000000 x0000
  x30FD
          000000000000000 x0000
  x30FE
          000000000000000 x0000
  x30FF
          000000000000000 x0000
  = x3100
          000000000000000 x0000
  x3101
  x3102
          000000000000000 x0000
```

Dopo istruzione 4

AND R2, R2, x0000 ; azzera R2										
	RO	x0000	0	R4	x0000	0	PC	x30FA	12538	
	R1	x30F4	12532	R5	x0000	0	IR	x54A0	21664	
	R2	x0000	0	R6	x0000	0	CC	Z		
	R3	x0000	0	R7	x0000	0				
	x30F4	00110	0010000001) x:	3102					
	x30F5	00000	00000000000) x	0000					
	x30F6	11100	0101111010) x1	E2F4		LEA	R1, x3	0F4	
	x30F7	00010	1000110111) x.	146E		ADD	R2, R1	, x000E	
	x30F8	00110	1001111010) x	34F4		ST	R2, x3	0F4	
	x30F9	01010	1001010000) x	54A0		AND	R2, R2	, x0000	
-	x30FA	00010	1001010010.	l x.	14A5		ADD	R2, R2	, x0005	
	x30FB	01110	1000100111) x	744E		STR	R2, R1	x000E	
	x30FC	10100	1101111010) x,	A6F4		LDI	R3, x3	0F4	
	x30FD	00000	00000000000) x	0000					
	x30FE	00000	0000000000) x	0000					
	x30FF	00000	00000000000) x	0000					
	x3100	00000	00000000000) x	0000					
	x3101	00000	00000000000) x	0000					
	x3102	00000	00000000000) x	0000					

Dopo istruzione 5

```
ADD R2, R2, x0005
                          ; inizializza R2 a 5
                                            PC x30FB 12539
      R0 x0000 0
                         R4 x0000 0
     R1 x30F4 12532
                         R5 x0000 0
                                            IR x14A5 5285
                                            CC P
     R2 x0005 5
                         R6 x0000 0
     R3 x0000 0
                         R7 x0000 0
  x30F4 0011000100000010 x3102
  x30F5 000000000000000 x0000
                                                R1, x30F4
  x30F6 1110001011110100 xE2F4
                                          LEA
  x30F7 0001010001101110 x146E
                                          ADD
                                                R2, R1, x000E
  = x30F8 0011010011110100
                          x34F4
                                          ST
                                                R2, x30F4
                          x54A0
                                                R2, R2, x0000
  x30F9 0101010010100000
                                          AND
  x30FA 0001010010100101
                                                R2, R2, x0005
                                                R2, R1 x000E
  → x30FB 0111010001001110 x744E
                                          STR
   x30FC
          1010011011110100
                          xA6F4
                                          LDI
                                                R3, x30F4
   x30FD 0000000000000000
                          x0000
   x30FE 0000000000000000 x0000
  x30FF 0000000000000000
                          x0000
   x3100
          0000000000000000
                          x0000
          00000000000000000
                          x0000
  x3101
  x3102
          00000000000000000
                          x0000
```

Dopo istruzione 6

```
STR R2, R1, x000E ; scrive R2 in M(R1+14=3102)
     R0 x0000 0
                         R4 x0000 0
                                            PC x30FC 12540
                                            IR x744E 29774
     R1 x30F4 12532
                        R5 x0000 0
                        R6 x0000 0
     R2 x0005 5
     R3 x0000 0
                        R7 x0000 0
  x30F4 0011000100000010 x3102
  x30F5 000000000000000 x0000
  x30F6 1110001011110100 xE2F4
                                         LEA
                                               R1, x30F4
                                               R2, R1, x000E
  x30F7
          0001010001101110 x146E
                                         ADD
  x30F8 0011010011110100 x34F4
                                         ST
                                               R2, x30F4
  x30F9 0101010010100000 x54A0
                                         AND
                                               R2, R2, x0000
  x30FA 0001010010100101 x14A5
                                         ADD
                                               R2, R2, x0005
  x30FB 0111010001001110 x744E
                                         STR
                                               R2, R1 x000E
  → x30FC 1010011011110100
                          xA6F4
                                         LDI
                                               R3, x30F4
  x30FD 000000000000000 x0000
  x30FE 000000000000000 x0000
  x30FF
          00000000000000000
                          x0000
  x3100 000000000000000 x0000
  x3101 000000000000000 x0000
  x3102 000000000000101 x0005
```

Dopo istruzione 7

```
LDI R3, x30F4; scrive M(M(30F4))=5 in R3
```

```
R4 x0000 0
                                            PC x30FD 12541
   RO XOOOO O
   R1 x30F4 12532
                       R5 x0000
                                 0
                                            IR xA6F4 -22796
   R2 x0005 5
                       R6 x0000 0
                                            CC P
   R3 x0005 5
                       R7 x0000 0
x30F4 0011000100000010 x3102
x30F5 000000000000000 x0000
x30F6 1110001011110100 xE2F4
                                         LEA
                                               R1, x30F4
x30F7
        0001010001101110 x146E
                                         ADD
                                               R2, R1, x000E
x30F8
        0011010011110100
                         x34F4
                                         ST
                                               R2, x30F4
                                               R2, R2, x0000
x30F9 0101010010100000 x54A0
                                         AND
x30FA 0001010010100101 x14A5
                                         ADD
                                               R2, R2, x0005
x30FB
        0111010001001110
                                               R2, R1 x000E
                         x744E
                                         STR
 x30FC
        1010011011110100
                                               R3, x30F4
                                         LDI
→ x30FD
        00000000000000000
        000000000000000 x0000
 x30FE
        00000000000000000
                         x0000
 x30FF
        0000000000000000000
 x3100
        00000000000000000
x3101
                         x0000
 x3102
        00000000000000101
                         x0005
```

Esempio sulle istruzioni di controllo

- ; programma che somma un vettore di numeri in memoria
- ; la somma termina appena si incontra un valore nullo
- ; programma situato in memoria a partire da x3000
- ; vettore situato in memoria a partire da x3008

loop	LEA AND LDR BRZ	R0, table R2, R2, #0 R1, R0, #0 finish	; carica 3008 in puntatore ; azzera totalizzatore ; legge prossimo numero ; se nullo ha finito
	ADD	R2, R2, R1	;somma a totalizzatore
	ADD	RO, RO, #1	;incrementa puntatore
	BRNZP	loop	;prossimo numero
finish	ST	R2, result	; scrive risultato in memoria
table			;vettore di numeri
result			;risultato

CPU e memoria all'inizio del programma

```
R0 x0000 0
                      R4 x0000 0
                                          PC x3000 12288
   R1 x0000 0
                      R5 x0000 0
                                          IR x0000 0
   R2 x0000 0
                      R6 x0000 0
   R3 x0000 0
                      R7 x0000 0
→ x3000 1110000000001000 xE008
                                       LEA
                                              RO, table
 x3001 0101010010100000 x54A0
                                       AND
                                              R2, R2, x0000
x3002 0110001000000000 x6200 loop
                                       LDR
                                              R1, R0 x0000
x3003 0000010000000111 x0407
                                       BRZ
                                              finish
= x3004 0001010010000001 x1481
                                       ADD
                                              R2, R2, R1
                                              RO, RO, x0001
= x3005 0001000000100001 x1021
                                       ADD
x3006 0000111000000010 x0E02
                                       BRNZP loop
x3007 0011010000001101 x340D finish
                                       ST
                                             R2, result
x3008 0000000000000110
                       x0006
                              table
x3009 000000000010100 x0014
x300A 1110010000000000 xE400
x300B 0111100000000000 x7800
        000000000000000 x0000
x300C
 x300D 000000000000000 x0000
                              result
```

Dopo la lettura del primo numero R4 x0000 0 PC x3003 12291 R0 x3008 12296 6 R5 x0000 IR x6200 25088 R1 x0006 R2 x0000 0 R6 x0000 0 CC P x0000 0 R7 x0000 0 x3000 111000000001000 xE008 LEA RO, table x3001 0101010010100000 x54A0 AND R2, R2, x0000 R1, R0 x0000 LDR x3002 0110001000000000 x6200 → x3003 0000010000000111 x0407 BRZ finish R2, R2, R1 00010100100000001 ADDx3004 x1481 x3005 0001000000100001 x1021 ADDRO, RO, x0001 BRNZP loop x3006 0000111000000010 x0E02 0011010000001101 R2, result x3007 x340D finish x3008 0000000000000110 x0006 table x3009 0000000000010100 x300A 1110010000000000 xE400 x7800 x300B 01111000000000000 x0000 x300C 000000000000000000 = x300D 000000000000000 x0000 result

Dopo la prima esecuzione del ciclo RO x3009 12297 R4 x0000 0 PC x3002 12290 R1 x0006 6 R5 x0000 0 IR x0E02 3586 R2 x0006 6 R6 x0000 0 R3 x0000 0 R7 x0000 0 RO, table x3000 111000000001000 xE008 LEA x3001 0101010010100000 x54A0 AND R2, R2, x0000 → x3002 0110001000000000 x6200 loop R1, R0 x0000 LDR x3003 0000010000000111 x0407 BRZ finish x3004 0001010010000001 x1481 ADD R2, R2, R1 x3005 0001000000100001 x1021 ADD RO, RO, x0001 BRNZP loop x3006 0000111000000010 x0E02 x3007 0011010000001101 x340D finish ST R2, result x3008 0000000000000110 x0006 table x3009 0000000000010100 x0014 x300A 11100100000000000 xE400 = x300B 0111100000000000 x7800 00000000000000000 x300C x0000 000000000000000 x0000 x300D result

Termine del programma

```
RO x300C 12300
                      R4 x0000 0
                                          PC x3008 12296
   R1 x0000 0
R2 x5C1A 23578
                                          IR x340D 13325
                      R5 x0000 0
                                         CC Z
                      R6 x0000 0
                     R7 x0000 0
   R3 x0000 0
x3000 1110000000001000 xE008
                                       LEA
                                             RO, table
x3001 0101010010100000 x54A0
                                       AND
                                             R2, R2, x0000
x3002 0110001000000000 x6200 loop
                                       LDR
                                             R1, R0 x0000
x3003 0000010000000111 x0407
                                       BRZ
                                             finish
x3004 0001010010000001 x1481
                                       ADD
                                             R2, R2, R1
x3005 0001000000100001 x1021
                                       ADD
                                             RO, RO, x0001
                                       BRNZP loop
x3006 0000111000000010 x0E02
       00110100000001101
                                             R2, result
 x3007
                       x340D
                              finish
→ x3008 0000000000000110 x0006
                              table
 x3009 000000000010100 x0014
x300A 1110010000000000 xE400
 x300B
        0111100000000000 x7800
x300C 000000000000000 x0000
x300D 0101110000011010 x5ClA result
```

In sintesi...

- Abbiamo visto due esempi di esecuzione di programmi in linguaggio macchina.
- Il comportamento dei due programmi è stato discusso mostrando il contenuto della CPU LC-2 e della sua memoria di lavoro.
- Abbiamo dato una rappresentazione simbolica dei programmi (con i nomi delle istruzioni LC-2 invece della loro codifica binaria).
- Abbiamo adottato sia un'analisi passo passo sia un'analisi per blocchi di istruzioni.

Vedremo gli strumenti software che ci consentono queste attività.

