

Problema 2.14

Scrivere un programma **SEC** per porre **0** in tutte le locazioni di memoria da 0100₈ a 0200₈.

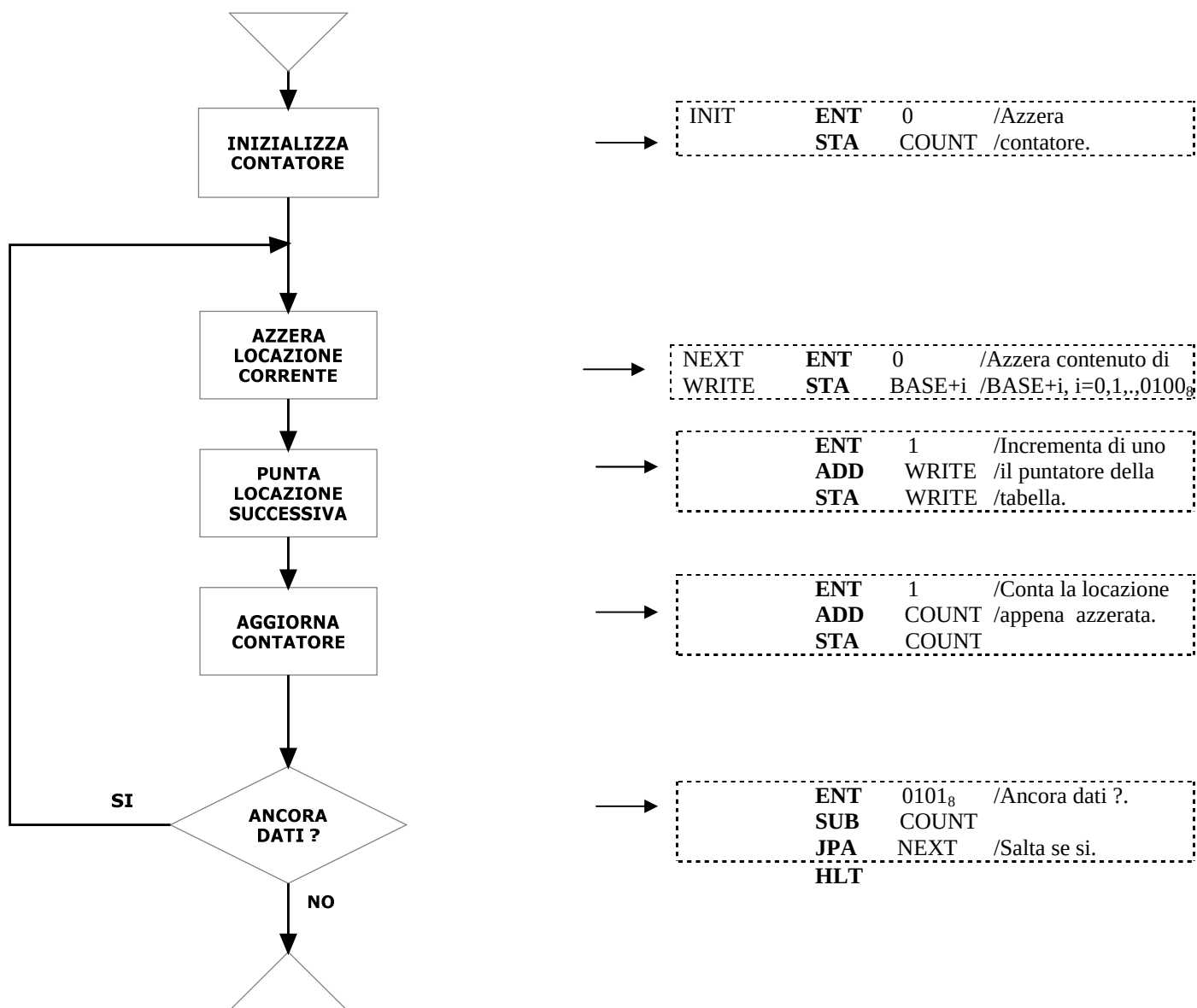
Soluzione

L'algoritmo esposto produce l'azzeramento del contenuto delle locazioni, ad iniziare da quella ad indirizzo più basso, contando le $0200_8 - 0100_8 + 1 = 0101_8$ locazioni interessate. Per fare questo l'istruzione che scrive (**STA** , 0 , 0100) viene iterativamente modificata aggiornando il puntatore **0100₈** di una unità dopo ogni scrittura in memoria.

In prima stesura il programma si scrive usando un linguaggio simbolico per cui si assegna il nome **BASE** all'indirizzo della prima locazione della quale si vuole azzerare il contenuto.

Si sceglie, inoltre di memorizzare la variabile di conteggio nella locazione di indirizzo 0300₈, alla quale si assegna il nome **COUNT**.

Fatte queste premesse si riporta il flow chart e la scrittura simbolica del programma.



Supposto di memorizzare il programma nella memoria del SEC, a partire dall'indirizzo 0000 e di tradurre gli indirizzi di ogni istruzione (mantenendo i simboli mnemonici per il solo codice operativo), la routine può essere riscritta nel seguente modo:

0000	ENT	,0,	0000	/Azzera contatore
0001	STA	,0,	0300	
0002	ENT	,0,	0000	/Azzera locazione corrente
0003	STA	,0,	0100	
0004	ENT	,0,	0001	/Punta alla locazione successiva
0005	ADD	,0,	0003	
0006	STA	,0,	0003	
0007	ENT	,0,	0001	/Conta locazione appena azzerata
0010	ADD	,0,	0300	
0011	STA	,0,	0300	
0012	ENT	,0,	0101	/Verifica se ci sono ancora dati
0013	SUB	,0,	0300	
0014	JPA	,0,	0002	/Ripeti ciclo se si
0015	HLT	,0,	0000	

I valori dei simboli sono evidentemente (in base ottale):

INIT	=	0000
COUNT	=	0300
NEXT	=	0002
WRITE	=	0003
BASE	=	0100