

7.21 Subroutine

Quando un insieme di istruzioni viene ripetuto più volte all'interno di un programma, conviene definirlo come **sottoprogramma** o subroutine.

Utilizzando le subroutine si hanno due vantaggi importanti, innanzitutto un risparmio di memoria di programma in quanto si eliminano ripetizioni inutili e in secondo luogo si ottiene un'organizzazione modulare del programma che ne facilita la lettura e la manutenzione.

Nel PLC CJ1 si hanno a disposizione 1024 subroutine, divisibili in **locali** (v. fig. 7.171), richiamabili cioè solo dal task (processo) in cui sono state create e in **globali** richiamabili invece da tutti i task del programma.

Le subroutine locali devono essere inserite dopo l'ultima istruzione del task e prima dell'istruzione END (001).

Ciascuna subroutine deve essere compresa tra le istruzioni SBN (092) e RET (093), e può essere richiamata tramite l'istruzione SBS (091).

In particolare con l'istruzione SBN (092) (Subroutine Number) viene definito l'inizio di una subroutine e specificato il numero compreso tra 0 e 1023; con l'istruzione RET (093) (Return) viene definita la fine di una subroutine.

Infine, con l'istruzione SBS (091) (Subroutine Start) viene definita la chiamata ad una subroutine specificata.

Se necessario una subroutine può essere richiamata più volte, utilizzando l'istruzione SBS in più punti del programma all'interno del task, da notare che una subroutine non può richiamare se stessa.

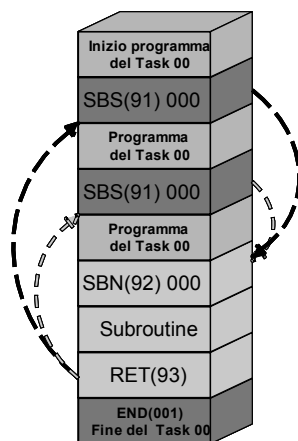


Fig. 7.172 - Funzionamento delle subroutine locali.

Nell'esempio di programmazione mostrato nella fig. 7.173, all'interno del task ciclico 00, vengono inserite le condizioni di chiamata alle subroutine locali 1 e 2.

Successivamente, prima dell'istruzione END (001), vengono definite le due subroutine.

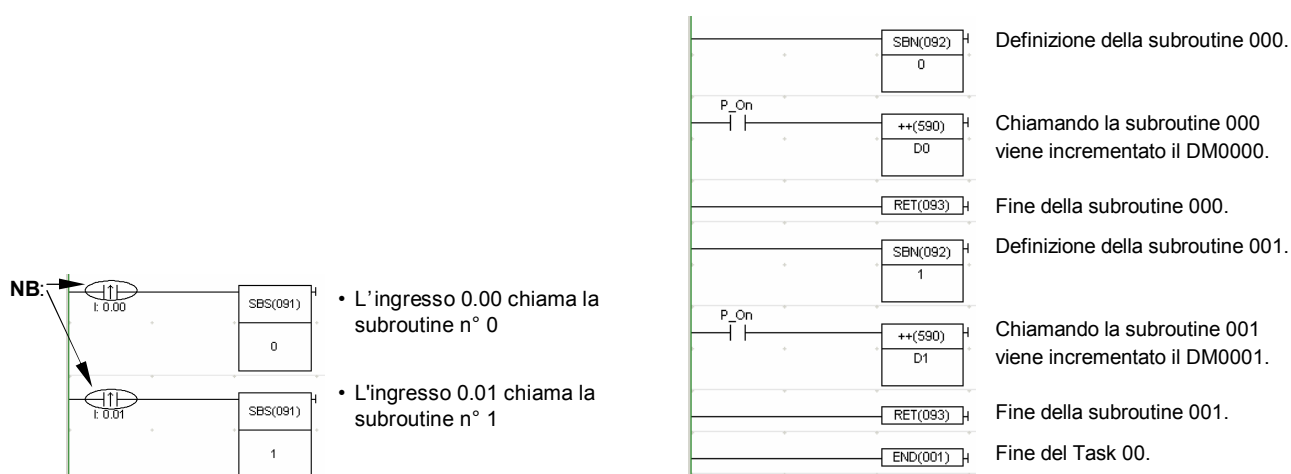



Fig. 7.173 - Esempio di applicazione delle subroutine locali.

Le subroutine globali devono essere definite all'interno del task ad interrupt 00.

Per creare il task ad interrupt 00 occorre utilizzare il software CX-Programmer.

Per aggiungere un nuovo task al programma, occorre fare clic sull'icona  nella barra degli strumenti.

A questo punto, fare clic con il tasto destro del mouse sul nuovo task e selezionare **Proprietà** per accedere alla finestra di impostazioni del task.

Infine, selezionare come tipo di task **Task di interrupt 00** e premere il tasto [Invio] sulla tastiera per confermare, come mostrato nella fig. 7.174a.

Ciascuna subroutine globale (v. fig. 7.174b) deve essere compresa tra le istruzioni GSBN (751) e GRET (752) e può essere richiamata da qualsiasi altro task tramite l'istruzione GSBS (750).

In particolare, l'istruzione GSBN (092) (Global Subroutine Number) consente di definire l'inizio di una subroutine globale e ne specifica il numero compreso tra 0 e 1023, mentre l'istruzione GRET (093) (Global Subroutine Return) permette di definire la fine di una subroutine; infine GSBS (091) (Global Subroutine Start) consente di definire la chiamata alla subroutine specificata, come mostrato nella fig. 4.74b.

La stessa subroutine globale può essere richiamata da task differenti utilizzando l'istruzione GSBS (750) relativa a quella subroutine.

Da notare che è necessario non assegnare ad una subroutine locale un numero già riservato ad una subroutine globale e viceversa. In caso contrario CX-Programmer genererà un errore in fase di compilazione e non sarà possibile scaricare il programma nel PLC.

Nell'esempio mostrato nella fig. 7.174c all'interno del task ad interrupt 00 viene definita la subroutine globale 3, mediante l'istruzione GSBS (750) è possibile richiamare questa subroutine da tutti gli altri task, compresi quelli ad interrupt.

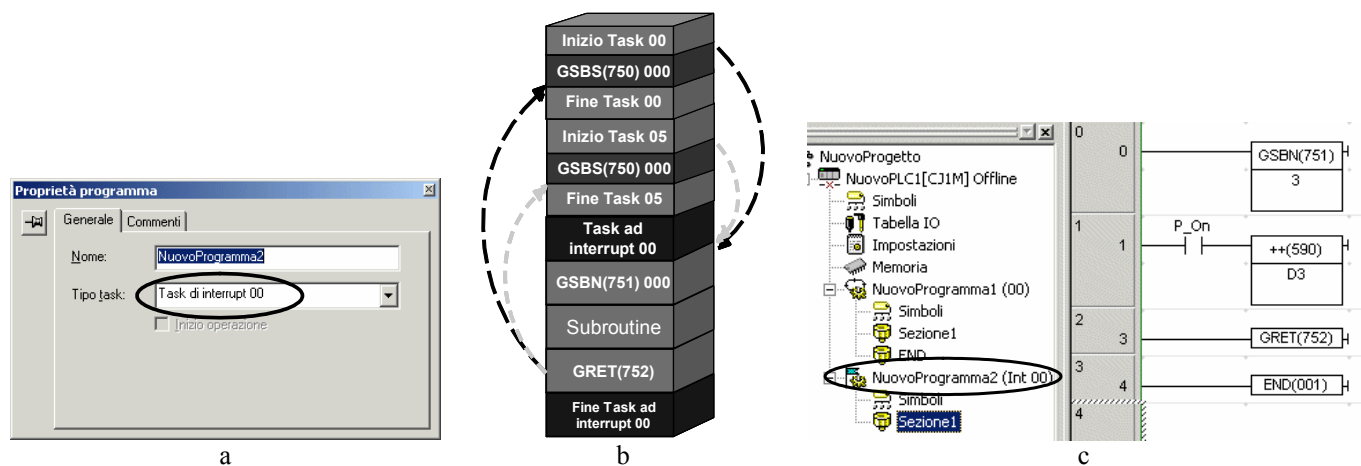


Fig. 7.174 - a) Creazione di un task ad interrupt 00 - b) Funzionamento delle subroutine globali - c) Esempio di applicazione delle subroutine globali.