

1. FLAG DI CONTROLLO DELLO STATO SFC

1.1. Descrizione generale

Con la versione LogicLab 3.14.9 sono disponibili alcuni flag per il controllo dello stato di una POU in linguaggio SFC.

I flag in questione sono i seguenti:

- **<nome POU>_HOLD_SFC** (type BOOL)
- **<nome POU>_RESET_SFC** (type BOOL)

dove <nome POU> indica il nome del blocco programmabile (PROGRAM o FUNCTION_BLOCK).

Ad esempio, se il blocco programmabile è un PROGRAM in linguaggio SFC di nome Main, i flag di controllo del PROGRAM Main si chiameranno rispettivamente: Main_HOLD_SFC e Main_RESET_SFC.

Esiste un coppia di flag di controllo anche per tutte le ACTION SFC contenute a loro volta in una POU SFC. Se ad esempio nel PROGRAM Main di cui sopra si avesse una ACTION realizzata anch'essa in SFC ed avente nome Execute, i flag di controllo di questa ACTION sarebbero rispettivamente: Main_Execute_HOLD_SFC e Main_Execute_RESET_SFC.

La funzione dei flag è descritta nei paragrafi che seguono.

1.2. Flag <nome POU>_HOLD_SFC

Il flag <>_HOLD_SFC ha le seguenti caratteristiche:

- il suo valore di default è FALSE
- quando il valore del flag viene posto a TRUE, il blocco SFC a cui si riferisce (quello avente nome uguale a <nome POU>) viene mantenuto nello stato corrente (hold) e nessuna parte del codice del blocco viene eseguita
- quando il valore del flag viene riportato a FALSE, l'esecuzione del blocco SFC riprende esattamente dal punto in cui si trovava al momento in cui era stato posto in HOLD mediante <>_HOLD_SFC := TRUE

1.3. Flag <nome POU>_RESET_SFC

Il flag <>_RESET_SFC ha le seguenti caratteristiche:

- il suo valore di default è FALSE
- quando il valore del flag viene posto a TRUE, il blocco SFC a cui si riferisce (quello avente nome uguale a <nome POU>) viene riportato allo stato iniziale ovvero allo stato di esecuzione della ACTION INIT.
- il flag <>_RESET_SFC è di tipo auto-reset il che significa che se viene posto a TRUE da programmatore il suo stato ritorna automaticamente a FALSE dopo che la sua azione di reset è stata eseguita

- non è quindi necessario riportare a FALSE lo stato di <>_RESET_SFC

1.4. Visibilità dei flag

I flag _HOLD_SFC e _RESET_SFC sono generati automaticamente dal compilatore e fanno parte delle **variabili locali** della POU in questione.

L'ambiente di sviluppo **non visualizza** i suddetti flag nell'elenco variabili delle POU. I flag di controllo sono quindi delle variabili nascoste ma comunque utilizzabili all'interno del codice.

1.5. Controllo dall'esterno di un FUNCTION_BLOCK SFC

Per poter consentire il controllo dall'esterno di un programma o di un blocco funzionale SFC è stata implementata la seguente funzionalità:

- il compilatore genera automaticamente i flag <>_RESET_SFC e <>_HOLD_SFC
- nel caso di **FUNCTION_BLOCK** il programmatore ha comunque la facoltà di dichiarare tra le **VAR_INPUT** due flag aventi esattamente i nomi dei flag di controllo SFC della POU e il tipo di dato BOOL
- nel caso di **PROGRAM** il programmatore ha comunque la facoltà di dichiarare tra le **VAR_GLOBAL** due flag aventi esattamente i nomi dei flag di controllo SFC della POU e il tipo di dato BOOL
- nel caso di dichiarazione esplicita di cui sopra, il compilatore utilizzerà le variabili dichiarate tra le **VAR_INPUT** o le **VAR_GLOBAL** e non quelle generate in automatico (che quindi non saranno generate).

In questo modo sarà possibile controllare lo stato di funzionamento dello schema SFC dall'esterno, utilizzando le variabili di input del blocco.

Es:

```
FUNCTION_BLOCK Prova
  VAR_INPUT
    ...
    Prova_RESET_SFC : BOOL; (* Control flag explicitly declared *)
  END_VAR
  ...
END_FUNCTION_BLOCK

PROGRAM Main
  VAR
    ...
    blocco : Prova; (* SFC block instance *)
  END_VAR
  ...
  (* Reset SFC block state *)
  blocco.Prova_RESET_SFC := TRUE;
  ...
END_PROGRAM
```

1.6. Libreria macro SFC

È disponibile una libreria (SFCControl.pll) che consente di gestire gli stati SFC mediante dei comandi anziché delle impostazioni di variabili.

La libreria in oggetto è composta da macro utilizzabili nel solo linguaggio ST.

1.7. Utilizzi dei flag di controllo

Seguono alcuni esempi di utilizzo dei flag di controllo SFC. Si assume che la POU SFC in questione si chiami Main.

- Hold (freeze) di SFC:

```
Main_HOLD_SFC := TRUE;
```

- Ripartenza dallo stato di hold:

```
Main_HOLD_SFC := FALSE;
```

- Ripartenza dallo stato iniziale di un SFC in hold:

```
Main_RESET_SFC := TRUE;  
Main_HOLD_SFC := FALSE;
```

- Reset allo stato iniziale e ripartenza immediata di un SFC:

```
Main_RESET_SFC := TRUE; (* Reset automatico dal compilatore *)
```