

VARIABLES

VAR\_GLOBAL

Fotocellula AT %IX0.0 : BOOL; (\* Ingresso fotocellula \*)

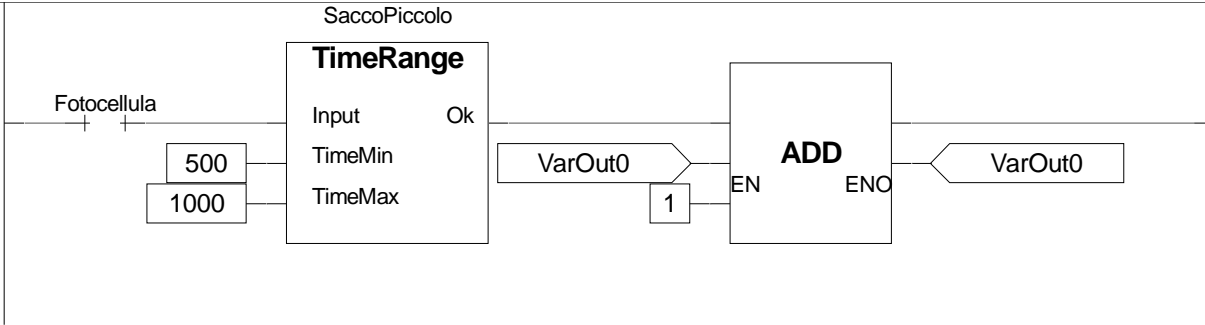
END\_VAR

	Project : BagCounter	
	VARIABLES :	
	Release :	Ver :1.00
	Author :	Date:20/03/2012
	Note :	Page:1 of 1

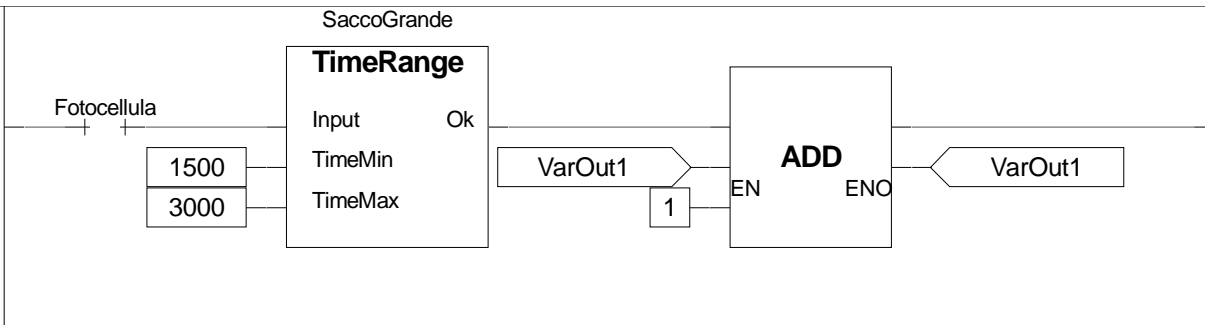
```

VAR
SaccoPiccolo : TimeRange; (* FB controllo sacco piccolo *)
SaccoGrande : TimeRange; (* FB controllo sacco grande *)
END_VAR
    
```

1



2



Project : BagCounter	
PROGRAM : BagCounter	
Release :	Ver :1.00
Author :	Date:20/03/2012
Note :	Page:1 of 1

PROGRAM StartUp

```

1 (* ***** *)
2 (* DEFINIZIONE ETICHETTE SIMULATORE *)
3 (* ***** *)
4 (* Eseguo definizione etichette sul simulatore. *)
5
6   CfgLogInp0Descr:='Fotocellula'; (* Definizione etichetta ingresso LI 0 *)
7
8   CfgVarOut0Min:=0.0; (* Minimo valore variabile VarO 0 *)
9   CfgVarOut0Max:=9999.9; (* Massimo valore variabile VarO 0 *)
10  CfgVarOut0Descr:='Piccolo'; (* Definizione etichetta variabile VarO 0 *)
11
12  CfgVarOut1Min:=0.0; (* Minimo valore variabile VarO 1 *)
13  CfgVarOut1Max:=9999.9; (* Massimo valore variabile VarO 1 *)
14  CfgVarOut1Descr:='Grande'; (* Definizione etichetta variabile VarO 0 *)
15
16 (* [End of file] *)
17

```

	Project : BagCounter	
	PROGRAM : StartUp	
	Release : BagCounter	Ver :1.00
	Author :	Date:20/03/2012
	Note :	Page:1 of 1

```

VAR_INPUT
Input : BOOL; (* Input to detect *)
TimeMin : UINT; (* Minimum time (S) *)
TimeMax : UINT; (* Maximum time (S) *)
END_VAR

VAR_OUTPUT
Ok : BOOL; (* Time in range *)
END_VAR

VAR_EXTERNAL
SysTime : UDINT; (* System time (mS) *)
END_VAR

VAR
Pulse : BOOL; (* Pulse detection *)
TimeBf : UDINT; (* Buffer gestione tempo *)
TimeActive : UDINT; (* Tempo attivazione ingresso (mS) *)
END_VAR
    
```

```

1 (* ***** *)
2 (* FUNCTION BLOCK "TimeRange" *)
3 (* ***** *)
4 (* Questo blocco funzione controlla se l'ingresso rimane attivo per un tempo *)
5 (* compreso tra i valori "Min" e "Max". Se compreso si attiva per un loop *)
6 (* l'uscita di "Ok". *)
7 (* ----- *)
8 (* Eseguo reset uscite impulsive. *)
9
10 Ok:=FALSE; (* Time in range *)
11
12 (* Eseguo controllo se variazione stato ingresso. *)
13
14 IF (Input = Pulse) THEN RETURN; END_IF;
15 Pulse:=Input; (* Pulse detection *)
16
17 (* Controllo attivazione ingresso, su attivazione salvo tempo. *)
18
19 IF (Input) THEN
20     TimeBf:=SysTime; (* Buffer gestione tempo *)
21 END_IF;
22
23 (* Controllo disattivazione ingresso, su disattivazione controllo *)
24 (* tempo trascorso e gestisco counter relativo. *)
25
26 IF NOT(Input) THEN
27     TimeActive:=SysTime-TimeBf; (* Tempo attivazione ingresso (mS) *)
28
29     (* Controllo se tempo di attivazione compreso tra valori di soglia. *)
30
31     IF (TimeActive >= TimeMin) AND (TimeActive <= TimeMax)THEN
32         Ok:=TRUE; (* Time in range *)
33     END_IF;
34 END_IF;
35
36 (* [End of file] *)
37
    
```

Project : BagCounter	
FUNCTION BLOCK : TimeRange	
Release : BagCounter	Ver :1.00
Author :	Date:20/03/2012
Note :	Page:1 of 2

	Project : BagCounter	
	FUNCTION BLOCK : TimeRange	
	Release : BagCounter	Ver :1.00
	Author :	Date:20/03/2012
	Note :	Page:2 of 2