

VARIABLES

VAR\_GLOBAL

CPulse AT %IX255.0 : BOOL; (\* Impulso da contatore energia \*)

END\_VAR

	Project : InstantPower	
	VARIABLES :	
	Release : InstantPow	Ver :1.00
	Author :	Date:28/03/2013
	Note :	Page:1 of 1

PROGRAM ST

```

VAR
PrTimer : UDINT; (* Tempo arrivo impulso precedente (uS) *)
MyTimer : UDINT; (* Tempo arrivo impulso attuale (uS) *)
Aux : REAL; (* Ausiliario calcolo *)
PwConstant : REAL := 3600; (* Power constant *)
IPower : UDINT; (* Instant power *)
AuxPulse : BOOL; (* Ausiliario impulso *)
END_VAR

```

```

1 (* ***** *)
2 (* PROGRAMMA CALCOLO POTENZA Istantanea *)
3 (* ***** *)
4 (* Questo programma esegue il calcolo della potenza istantanea basandosi sul *)
5 (* tempo che intercorre tra due impulsi letti da un contatore di energia. *)
6 (* La formula è  $P=3600/(k*t)$  dove: *)
7 (* k= costante impulsi/kWh del contatore *)
8 (* t= tempo intercorso tra l'arrivo dell'impulso e il precedente. *)
9 (* ----- *)
10
11 (* ----- *)
12 (* CALCOLO POTENZA Istantanea *)
13 (* ----- *)
14 (* Eseguo gestione fronte arrivo impulso da contatore. *)
15
16 IF (CPulse = AuxPulse) THEN RETURN; END_IF;
17 AuxPulse:=CPulse; (* Ausiliario impulso *)
18 IF NOT(CPulse) THEN RETURN; END_IF;
19
20 (* Eseguo calcolo potenza istantanea. *)
21
22 MyTimer:=SysGetSysTime(TRUE); (* Tempo arrivo impulso attuale (uS) *)
23 Aux :=(PwConstant/1000.0)*TO_REAL(MyTimer-PrTimer)/1000000.0;
24 IPower:=TO_UDINT(3600.0/Aux);
25
26 (* Memorizzo tempo arrivo impulso attuale in precedente. *)
27
28 PrTimer:=MyTimer; (* Tempo arrivo impulso precedente (uS) *)
29
30 (* [End of file] *)
31

```

Project : InstantPower	
PROGRAM : ST	
Release : InstantPow	Ver :1.00
Author :	Date:28/03/2013
Note :	Page:1 of 1