

```

1  FUNCTION_BLOCK BarcodeRead
2
3  VAR_INPUT
4      Enable : BOOL; (* FB Enable *)
5      Fp : FILEP; (* File pointer *)
6  END_VAR
7
8  VAR_OUTPUT
9      ReadOk : BOOL; (* Fine lettura *)
10     Barcode : STRING[ 32 ]; (* Codice letto *)
11 END_VAR
12
13 VAR
14     CaseNr : USINT; (* Case gestione programma *)
15     Ch : USINT; (* Rx/Tx character *)
16     TimeBf : UDINT; (* Time buffer *)
17     i : UDINT; (* Auxiliary counter *)
18     Ptr : @USINT; (* Internal pointer *)
19 END_VAR
20
21 (* ***** *)
22 (* FUNCTION BLOCK "BarcodeRead" *)
23 (* ***** *)
24 (* Lettura codice barcode. *)
25 (* ----- *)
26
27 (* ----- *)
28 (* RESET BIT DI ONE SHOT *)
29 (* ----- *)
30 (* Eseguo reset bit di one shot. *)
31
32 ReadOk:=FALSE; (* Fine lettura *)
33
34 (* ----- *)
35 (* FB ENABLE *)
36 (* ----- *)
37 (* Check if the FB is enabled. *)
38
39 IF NOT(Enable) THEN CaseNr:=0; RETURN; END_IF;
40
41 (* ----- *)
42 (* CASES GESTIONE *)
43 (* ----- *)
44 (* Cases gestione programma. *)
45
46 CASE (CaseNr) OF
47
48     (* ----- *)
49     (* GESTIONE RICEZIONE FRAME RISPOSTA *)
50     (* ----- *)
51     (* Attesa ricezione caratteri. Controllo se ricevuto il carattere di *)
52     (* start frame "$BD". *)
53
54     0:
55     IF NOT(TO_BOOL(SysGetIChars(Fp))) THEN RETURN; END_IF;
56     TimeBf:=SysGetSysTime(TRUE); (* Time buffer *)
57     i:=Sysmemset(ADR(Barcode), 0, SIZEOF(Barcode));
58     Ptr:=ADR(Barcode); (* Internal pointer *)

```

```
59      CaseNr:=CaseNr+1; (* Case gestione programma *)
60
61      (* ----- *)
62      (* Ricezione stringa barcode, ricezione termina se non ricevo *)
63      (* caratteri per almeno 100 mS e su ricezione <CR>. *)
64
65      1:
66      IF ((SysGetSysTime(TRUE)-TimeBf) > 100000) THEN CaseNr:=CaseNr+1; RETURN;
        END_IF;
67
68      (* Eseguo loop ricezione caratteri trasmessi dal lettore. *)
69
70      WHILE (TO_BOOL(SysGetIChars(Fp))) DO
71          TimeBf:=SysGetSysTime(TRUE); (* Time buffer *)
72          Ch:=TO_USINT(Sysfgetc(Fp)); (* Rx/Tx character *)
73
74          (* Controllo se ricevuto <CR> e se ricevuto troppi caratteri. *)
75
76          IF (Ch = 16#0D) THEN CaseNr:=CaseNr+1; RETURN; END_IF;
77          IF (Ptr >= ADR(Barcode)+SIZEOF(Barcode)) THEN CaseNr:=0; RETURN; END_IF;
78
79          @Ptr:=Ch; (* Salvo dato letto *)
80          Ptr:=Ptr+1; (* Internal pointer *)
81      END_WHILE;
82
83      (* ----- *)
84      (* Attivo fine lettura. *)
85
86      2:
87      ReadOk:=TRUE; (* Fine lettura *)
88      CaseNr:=0; (* Case gestione programma *)
89      END_CASE;
90
91      (* [End of file] *)
92
93
```