



Structured Text Blocks



Engine_Invert

```

(* Engine Inverter in ST
   Copyright, 2023 Dogwood Valley Press

   Start/stop of operation
*)
IF (START_PB AND NOT Int_Reset) THEN
    Run := 1;
END_IF;
IF (Run AND NOT STOP_PB) THEN
    Run := 0;
END_IF;
(*
   Normal operation transition out of initial step
*)
IF Run AND (IStep = 0) THEN
    IStep := 1;
END_IF;
(*
   Transitions for normal operation
*)
CASE IStep OF
    1:
        IF PROX1 AND Run THEN
            IStep := 2;
        END_IF;
    2:
        Eng1_Tmr(IN:=1, PT:=2000);
        IF Eng1_Tmr.Q THEN
            Eng1_Tmr(IN:=0, PT:=2000); (* Must run timer again to reset *)
            IStep := 3;
        END_IF;
    3:
        IF PALL_UPLS AND Run THEN
            IStep := 4;
        END_IF;
    4:
        IF ROTR_DNLS AND Run THEN
            IStep := 5;
        END_IF;
    5:
        Clmp_Tmr(IN:=1, PT:=1500);
        IF Clmp_Tmr.Q THEN
            Clmp_Tmr(IN:=0, PT:=1000);
            IStep := 6;
        END_IF;
    6:
        IF ROTR_UPLS AND Run THEN
            IStep := 7;

```

```

        END_IF;
7:    IF ROTR_CWLS AND Run THEN
        IStep := 8;
    END_IF;
8:    IF ROTR_DNLS AND Run THEN
        IStep := 9;
    END_IF;
9:    UnClmp_Tmr(IN := 1, PT := 1000);
    IF UnClmp_Tmr.Q THEN
        UnClmp_Tmr(IN := 0, PT := 1000);
        IStep := 10;
    END_IF;
10:   IF ROTR_UPLS AND Run THEN
        IStep := 11;
    END_IF;
11:   IF ROTR_CCWLS AND Run THEN
        IStep := 12;
    END_IF;
12:   IF (NOT PALL_UPLS) AND Run THEN
        IStep := 13;
    END_IF;
13:   Eng2_Tmr(IN := 1, PT := 3000);
    IF Eng2_Tmr.Q THEN
        Eng2_Tmr(IN := 0, PT := 3000);
        IStep := 1;
    END_IF;
ELSE
    IStep := 0;
END_CASE;
(*
    Start/stop for reset operation
*)
IF (RESET_PB AND NOT Run) THEN
    Int_Reset := 1;
END_IF;
IF (Int_Reset AND RStep = 4) THEN
    Int_Reset := 0;
END_IF;
(*
    Reset initial transition and clear normal step number
*)
IF Int_Reset AND RStep = 0 THEN
    RStep := 1;
    IStep := 0;
END_IF;
(*
    Reset operation transitions
*)
CASE RStep OF
    1:
        RUnClmp_Tmr(IN := 1, PT := 1000);
        IF RUnClmp_Tmr.Q THEN
            RUnClmp_Tmr(IN := 0, PT := 1000);

```

```

        RStep := 2;
    END_IF;
2:
    IF ROTR_UPLS THEN
        RStep := 3;
    END_IF;
3:
    IF ROTR_CCWLS THEN
        RStep := 4;
    END_IF;
4:
    IF (NOT Int_Reset) THEN
        RStep := 0;
    END_IF;
ELSE
    RStep := 0;
END_CASE;
(*
    Control of outputs;
*)
IF (IStep = 2) THEN
    ENGL_RET := 1;
ELSE
    ENGL_RET := 0;
END_IF;
IF (IStep = 13) THEN
    ENG2_RET := 1;
ELSE
    ENG2_RET := 0;
END_IF;
IF ((IStep = 6) OR (IStep = 10) AND Run) OR (RStep = 2) THEN
    ROTR_UP := 1;
ELSE
    ROTR_UP := 0;
END_IF;
IF ((IStep = 4) OR (IStep = 8) AND Run) THEN
    ROTR_DOWN := 1;
ELSE
    ROTR_DOWN := 0;
END_IF;
IF ((IStep = 7) AND Run) THEN
    ROTAT_CW := 1;
ELSE
    ROTAT_CW := 0;
END_IF;
IF ((IStep = 11) AND Run) OR (RStep = 3) THEN
    ROTAT_CCW := 1;
ELSE
    ROTAT_CCW := 0;
END_IF;
IF (IStep >= 5) AND (IStep <= 8) THEN
    GRIP_CLOS := 1;
ELSE
    GRIP_CLOS := 0;
END_IF;
IF (IStep >= 3) AND (IStep <= 11) THEN
    PALL_UPCTL := 1;
ELSE
    PALL_UPCTL := 0;
END_IF;

```

