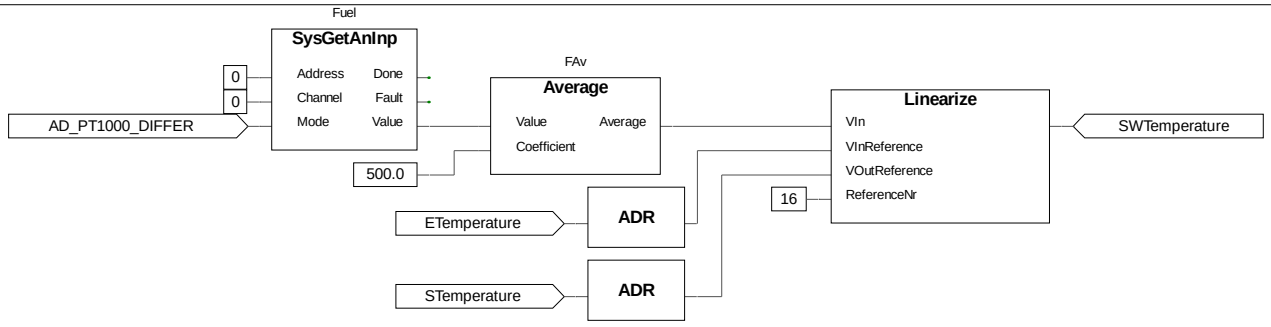
```
VAR_GLOBAL
ETemperature : ARRAY[ 0..15 ] OF REAL; (* External temperature (°C) *)
STemperature : ARRAY[ 0..15 ] OF REAL; (* Supply water temperature (°C) *)
LCurve AT %IX0.0 : BOOL; (* Low curve *)
MCurve AT %IX0.1 : BOOL; (* Medium curve *)
HCurve AT %IX0.2 : BOOL; (* High curve *)
END_VAR
```

PROGRAM SWaterTemp

```
VAR
Fuel : SysGetAnInp; (* Analog acquisition *)
SWTemperature : REAL; (* Supply water temperature (°C) *)
FAv : Average; (* Level average *)
END_VAR
```

1



	Project : ClimateCurve	
	PROGRAM : SWaterTemp	
	Release : ClimateCur	Ver :1.00
	Author :	Date:30/01/2014
	Note :	Page:1 of 1

PROGRAM Curves

```
VAR
i : USINT; (* Auxiliary counter *)
Curves : ARRAY[ 0..2 ] OF BOOL; (* Curve selectors *)
END_VAR
```

```
(* ********************************************************************** *)
(* PROGRAM "Curves"                                                       *)
(* ********************************************************************** *)
(* Questo programma esegue l'impostazione curve temperatura.              *)
(*                                                                        *)
(* ---------------------------------------------------------------------- *)

    (* ---------------------------------------------------------------- *)
    (* SET CLIMATE CURVES                                               *)
    (* ---------------------------------------------------------------- *)
    (* Su variazione selezione curva eseguo impostazione valori. *)

    IF (LCurve = Curves[0]) AND (MCurve = Curves[1]) AND (HCurve = Curves[2]) THEN RETURN; END_IF;

    (* Copio valori per one shot variazione. *)

    Curves[0]:=LCurve; (* Low curve *)
    Curves[1]:=MCurve; (* Low curve *)
    Curves[2]:=HCurve; (* Low curve *)

    (* Azzero curve di temperatura per inizializzare tutti i valori. *)

    FOR i:=0 TO SIZEOF(ETemperature)/4 DO ETemperature[i]:=0.0; END_FOR;
    FOR i:=0 TO SIZEOF(STemperature)/4 DO STemperature[i]:=0.0; END_FOR;

    (* ---------------------------------------------------------------- *)
    (* DEFINIZIONE CURVA ALTA                                           *)
    (* ---------------------------------------------------------------- *)
    (* Definizione valori primo segmento curva. *)

    IF (HCurve) THEN
        ETemperature[0]:=-5.0; (* External temperature (°C) *)
        STemperature[0]:=80.0; (* Supply water temperature (°C) *)

        ETemperature[1]:=-2.0; (* External temperature (°C) *)
        STemperature[1]:=83.0; (* Supply water temperature (°C) *)

        (* Definizione valori secondo segmento curva. *)

        ETemperature[2]:=0.0; (* External temperature (°C) *)
        STemperature[2]:=80.0; (* Supply water temperature (°C) *)

        ETemperature[3]:=15.0; (* External temperature (°C) *)
        STemperature[3]:=28.0; (* Supply water temperature (°C) *)

        (* Definizione valori terzo segmento curva. *)

        ETemperature[4]:=20.0; (* External temperature (°C) *)
        STemperature[4]:=25.0; (* Supply water temperature (°C) *)
    END_IF;

    (* ---------------------------------------------------------------- *)
```

```
    (* DEFINIZIONE CURVA MEDIA                                                *)
    (* ---------------------------------------------------------------------- *)
    (* Definizione valori primo segmento curva. *)

    IF (MCurve) THEN
        ETemperature[0]:=-5.0; (* External temperature (°C) *)
        STemperature[0]:=60.0; (* Supply water temperature (°C) *)

        (* Definizione valori secondo segmento curva. *)

        ETemperature[1]:=0.0; (* External temperature (°C) *)
        STemperature[1]:=60.0; (* Supply water temperature (°C) *)

        ETemperature[2]:=15.0; (* External temperature (°C) *)
        STemperature[2]:=43.0; (* Supply water temperature (°C) *)

        (* Definizione valori terzo segmento curva. *)

        ETemperature[3]:=20.0; (* External temperature (°C) *)
        STemperature[3]:=40.0; (* Supply water temperature (°C) *)
    END_IF;

    (* ---------------------------------------------------------------------- *)
    (* DEFINIZIONE CURVA BASSA                                                *)
    (* ---------------------------------------------------------------------- *)
    (* Definizione valori primo segmento curva. *)

    IF (LCurve) THEN
        ETemperature[0]:=-5.0; (* External temperature (°C) *)
        STemperature[0]:=40.0; (* Supply water temperature (°C) *)

        (* Definizione valori secondo segmento curva. *)

        ETemperature[1]:=0.0; (* External temperature (°C) *)
        STemperature[1]:=40.0; (* Supply water temperature (°C) *)

        ETemperature[2]:=15.0; (* External temperature (°C) *)
        STemperature[2]:=26.0; (* Supply water temperature (°C) *)

        (* Definizione valori terzo segmento curva. *)

        ETemperature[3]:=20.0; (* External temperature (°C) *)
        STemperature[3]:=25.0; (* Supply water temperature (°C) *)
    END_IF;

(* [End of file] *)
```

FUNCTION_BLOCK Average

(SFR054B220) Executes the average over a value
ENCRYPTED CODE

```
VAR_INPUT
Value : REAL; (* Value to mediate *)
Coefficient : REAL; (* Media coefficient *)
END_VAR

VAR_OUTPUT
Average : REAL; (* Average value *)
END_VAR
```

1

	Project : ClimateCurve	
	FUNCTION BLOCK : Average	
	Release : ClimateCur	Ver :1.00
	Author :	Date:30/01/2014
	Note :	Page:1 of 1

FUNCTION Linearize

(SFR054B220) Linearizes a nonlinear value
ENCRYPTED CODE

```
VAR_INPUT
VIn : REAL; (* Valore in ingresso *)
VInReference : @REAL; (* Valori di riferimento ingresso *)
VOutReference : @REAL; (* Valori di riferimento uscita *)
ReferenceNr : USINT; (* Numero valori di riferimento *)
END_VAR
```

1

	Project : ClimateCurve	
	FUNCTION : Linearize	
	Release : ClimateCur	Ver :1.00
	Author :	Date:30/01/2014
	Note :	Page:1 of 1