```
Machine Temp_Ctrl_Task1Impl

  TaskType: periodic(250)
  Priority: 5

  taskbody is
    tc1: Temp_Ctrl_Task1Impl.TCSense_Temperatures || Envir1Impl.ENSense_Temperatures;
    tc2: Temp_Ctrl_Task1Impl.TCCalculate_Average_Temperature;
    tc3: Temp_Ctrl_Task1Impl.TCDisplay_Current_Temperature || Envir1Impl.ENDisplay_Current_Temperature;
    tc4: Temp_Ctrl_Task1Impl.TCGet_Target_Temperature2 || Shared_Object1Impl.SOGet_Target_Temperature2;
    tc5: IF Temp_Ctrl_Task1Impl.TCTurnON_Heat_Source  END IF
         ELSE Temp_Ctrl_Task1Impl.TCTurnOFF_Heat_Source  END ELSE;
    tc6: Temp_Ctrl_Task1Impl.TCSet_Heat_Source_State || Shared_Object1Impl.SOSet_Heat_Source_State;
    tc7: Temp_Ctrl_Task1Impl.TCActuate_Heat_Source || Envir1Impl.ENActuate_Heat_Source;
    tc8: IF Temp_Ctrl_Task1Impl.TCSwitchOn_OverHeat_Alram  END IF
         ELSE Temp_Ctrl_Task1Impl.TCSwitchOff_OverHeat_Alram  END ELSE;
    tc9: Temp_Ctrl_Task1Impl.TCActuate_OverHeat_Alram || Envir1Impl.ENActuate_OverHeat_Alram

VARIABLES
  stm1
  stm2
  avt
  cttm2
  hsc
  ota

INVARIANTS
  stm1 ∈ ℤ
  avt ∈ ℤ
  stm2 ∈ ℤ
  hsc ∈ BOOL
  ota ∈ BOOL
  cttm2 ∈ ℤ

EVENTS
INITIALISATION ≙
 BEGIN
  THEN
   hsc ≔ FALSE
   ota ≔ FALSE
   avt ≔ 0
   cttm2 ≔ 0
   stm1 ≔ 0
   stm2 ≔ 0
 END

TCSense_Temperatures ≙
 Sensing
 BEGIN ANY
   actualIn t1
   actualIn t2
  WHEN
   t1 ∈ ℤ
   t2 ∈ ℤ
  THEN
   stm1 ≔ t1
   stm2 ≔ t2
 END

TCCalculate_Average_Temperature ≙
 ProcedureDef
 BEGIN
  THEN
   avt ≔ (stm1 + stm2) ÷ 2
 END

TCDisplay_Current_Temperature ≙
```

```
Actuating
BEGIN ANY
  actualOut tm_avt
 WHEN
  tm_avt = avt
  tm_avt ∈ ℤ
 THEN
   skip
END
```

TCGet_Target_Temperature2 ≜

```
ProcedureSynch
BEGIN ANY
  actualIn tm
 WHEN
  tm ∈ ℤ
 THEN
  cttm2 ≔ tm
END
```

TCTurnON_Heat_Source ≜

```
Branch
BEGIN
 WHEN
  avt < cttm2
 THEN
  hsc ≔ TRUE
END
```

TCTurnOFF_Heat_Source ≜

```
Branch
BEGIN
 WHEN
  avt ≥ cttm2
 THEN
  hsc ≔ FALSE
END
```

TCSet_Heat_Source_State ≜

```
ProcedureSynch
BEGIN ANY
  actualOut state
 WHEN
  state ∈ BOOL
  state = hsc
 THEN
   skip
END
```

TCActuate_Heat_Source ≜

```
Actuating
BEGIN ANY
  actualOut state_hsc
 WHEN
  state_hsc ∈ BOOL
  state_hsc = hsc
 THEN
   skip
END
```

TCSwitchOn_OverHeat_Alram ≜

```
Branch
BEGIN
 WHEN
  avt > Max
 THEN
  ota ≔ TRUE
END
```

TCSwitchOff_OverHeat_Alram ≜

```
Branch
BEGIN
 WHEN
  avt ≤ Max
 THEN
  ota ≔ FALSE
END

TCActuate_OverHeat_Alram ≜
Actuating
BEGIN ANY
  actualOut state_ota
 WHEN
  state_ota ∈ BOOL
  state_ota = ota
 THEN
   skip
END
```