CoreTemp Level-2 Telegram UDTs (TIA Portal)

INCLUDES A SET OF PLC DATA TYPES THAT IMPLEMENT A 1:1 MEMORY IMAGE OF THE **LEVEL-2 INDUSTRIAL TELEGRAMS** DESCRIBED IN THE CORETEMP MANUAL (INPUT TELEGRAM = DATA SENT FROM PLC TO CORETEMP, OUTPUT TELEGRAM = DATA SENT FROM CORETEMP TO PLC).

Objects provided (from the PLC point of view)

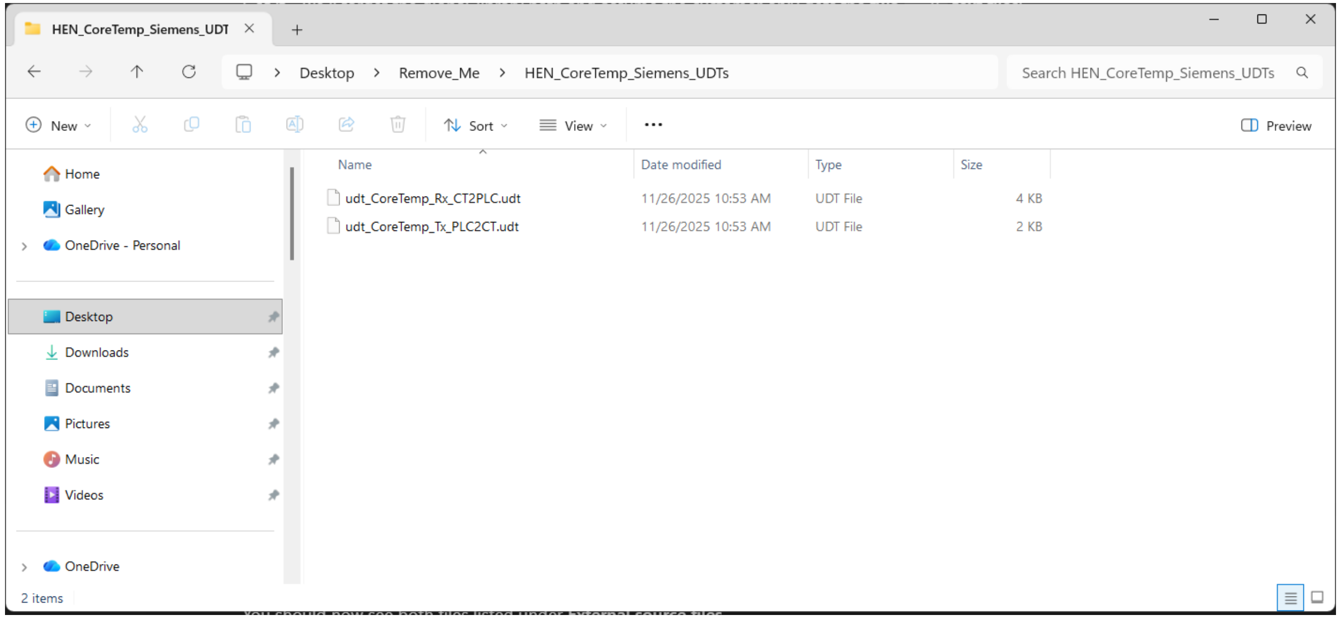
- **CoreTemp to PLC (Output telegram)**
  - udt_CoreTemp_Rx_CT2PLC
    - Status : udt_CT_CT2PLC_StatusBits
      - High-level status flags (Ready, measuring, end single measurement, end sequence, system lock, process complete, homing, etc.).
    - Error : udt_CT_CT2PLC_ErrorBits
      - Combined measurement and system error flags (cold/scrap measurement, lance blocked, homing error, safety/E-Stop, etc.).
    - Common : udt_CT_CT2PLC_Common
      - Heat number, total used wire, estimated tap temperature and bath level, level slope, etc.
    - Meas[0..5] : udt_CT_CT2PLC_Meas
      - Up to six recent measurements, each containing temperature, energy, bath level, weight, quality, type, used length and remaining coil length.

- **PLC to CoreTemp (Input telegram)**
  - udt_CoreTemp_Tx_PLC2CT
    - Status : udt_CT_PLC2CT_StatusBits
      - Command/control bits corresponding to the Level-2 status byte: system ready, start single measurement, start sequence, abort, reset, hot heel flags, etc.
    - Data : udt_CT_PLC2CT_Data
      - All non-bit fields: heat number, energy reference, tilt angle, Celox reference values, level offset, etc.

✓ The goal is to overlay the two UDTs directly onto the input and output address range of the CoreTemp communication module (Anybus / fieldbus interface). This avoids manual byte/word mapping.

Importing the CoreTemp Level-2 UDTs (TIA Portal v18)

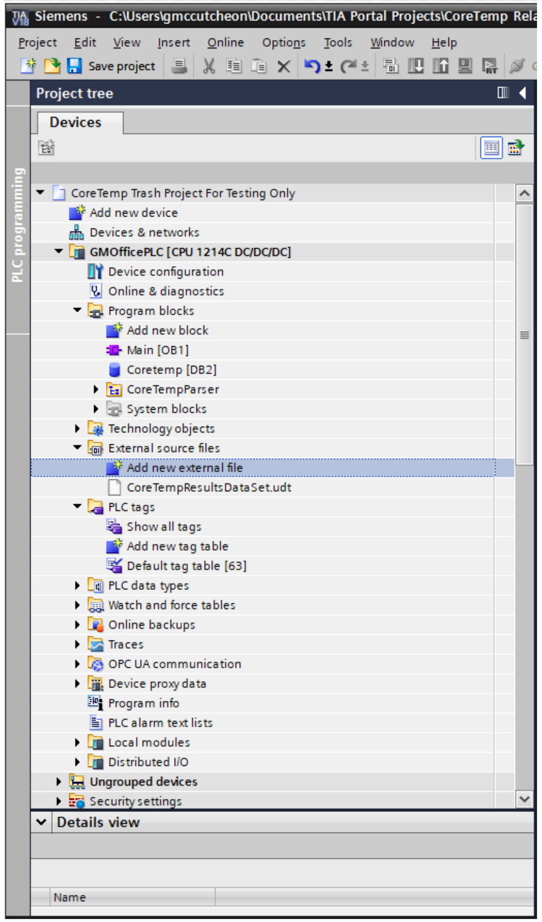
- The CoreTemp PLC data types are supplied as two plain-text source files:
  - udt_CoreTemp_Rx_CT2PLC.udt
    - Structure for the CoreTemp → PLC telegram (Output telegram).
  - udt_CoreTemp_Tx_PLC2CT.udt
    - Structure for the PLC → CoreTemp telegram (Input telegram).
- Each file contains the top-level UDT plus the nested UDTs (status bits, error bits, common data and measurement array)



Name	Date modified	Type	Size
udt_CoreTemp_Rx_CT2PLC.udt	11/26/2025 10:53 AM	UDT File	4 KB
udt_CoreTemp_Tx_PLC2CT.udt	11/26/2025 10:53 AM	UDT File	2 KB

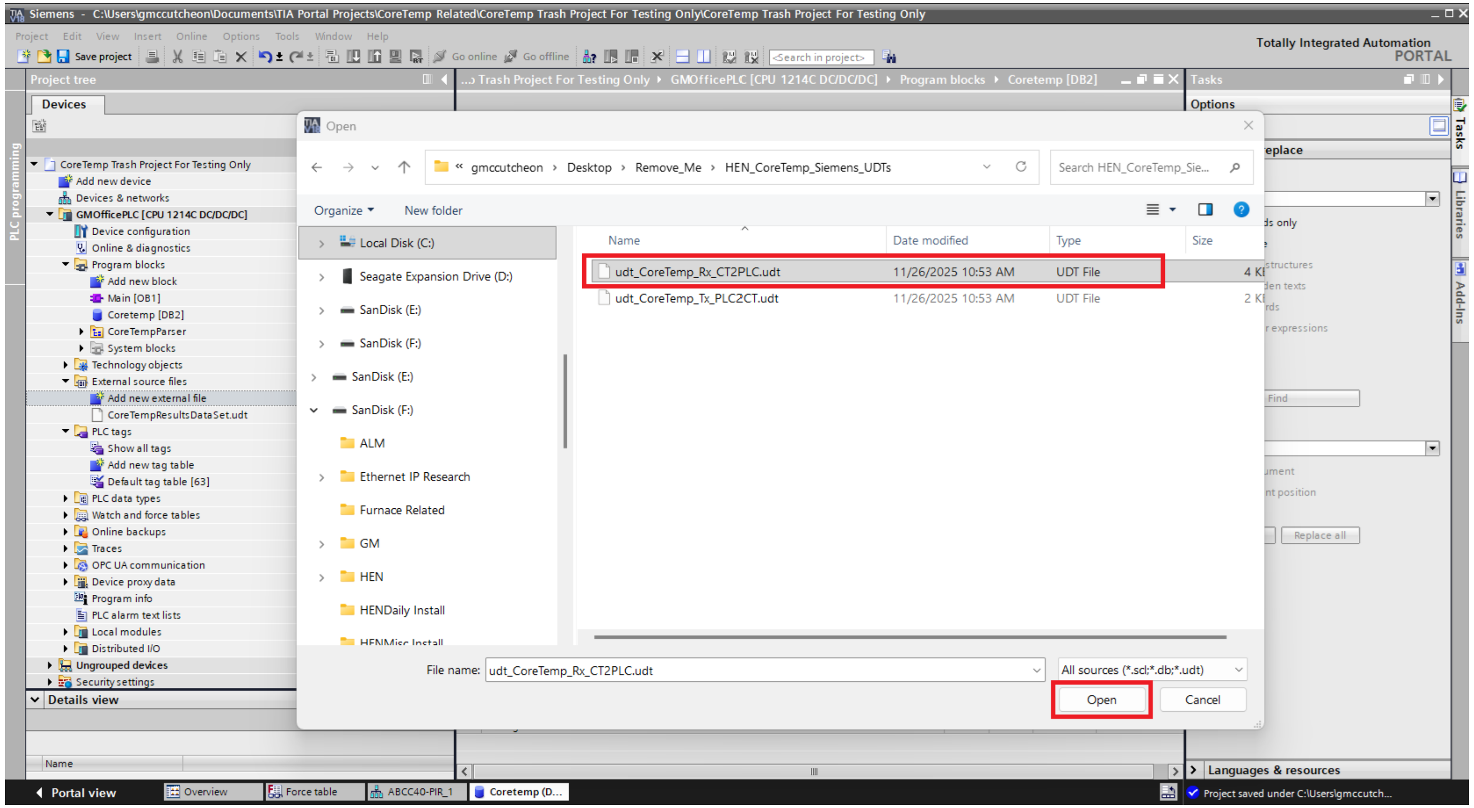
Add the .udt source files to the project

- In TIA Portal, open your project and select the CPU in the project tree.
- Expand External source files under the CPU.
- Click External source files → Add new external source



Add the udt_CoreTemp_Rx_CT2PLC.udt source files to the project

- Browse to udt_CoreTemp_Rx_CT2PLC.udt
- Select it and click Open.



Siemens - C:\Users\gmccutcheon\Documents\TIA Portal Projects\CoreTemp Related\CoreTemp Trash Project For Testing Only\CoreTemp Trash Project For Testing Only

...› Trash Project For Testing Only › GMOfficePLC [CPU 1214C DC/DC/DC] › Program blocks › Coretemp [DB2]

Open

« gmccutcheon › Desktop › Remove_Me › HEN_CoreTemp_Siemens_UDTs

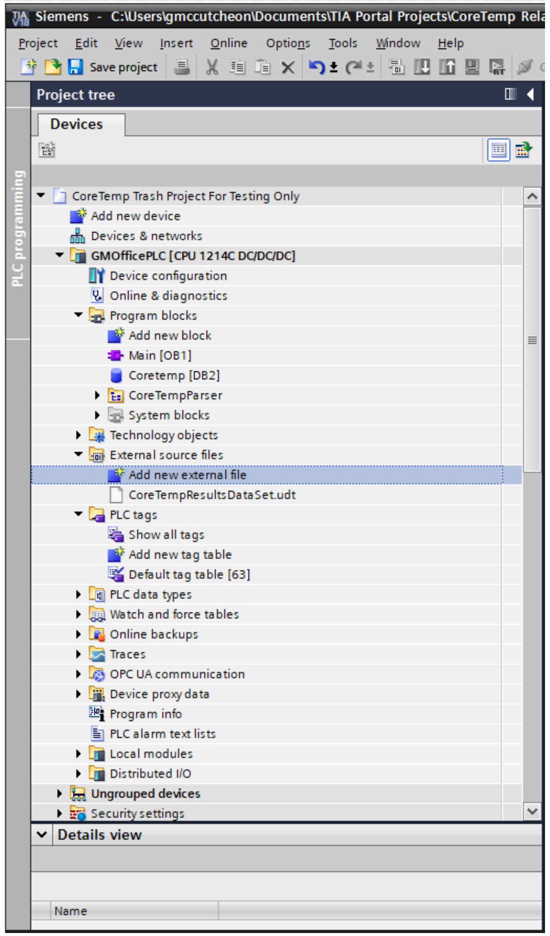
Name	Date modified	Type	Size
udt_CoreTemp_Rx_CT2PLC.udt	11/26/2025 10:53 AM	UDT File	4 KB
udt_CoreTemp_Tx_PLC2CT.udt	11/26/2025 10:53 AM	UDT File	2 KB

File name: udt_CoreTemp_Rx_CT2PLC.udt

All sources (*.scl;*.db;*.udt)

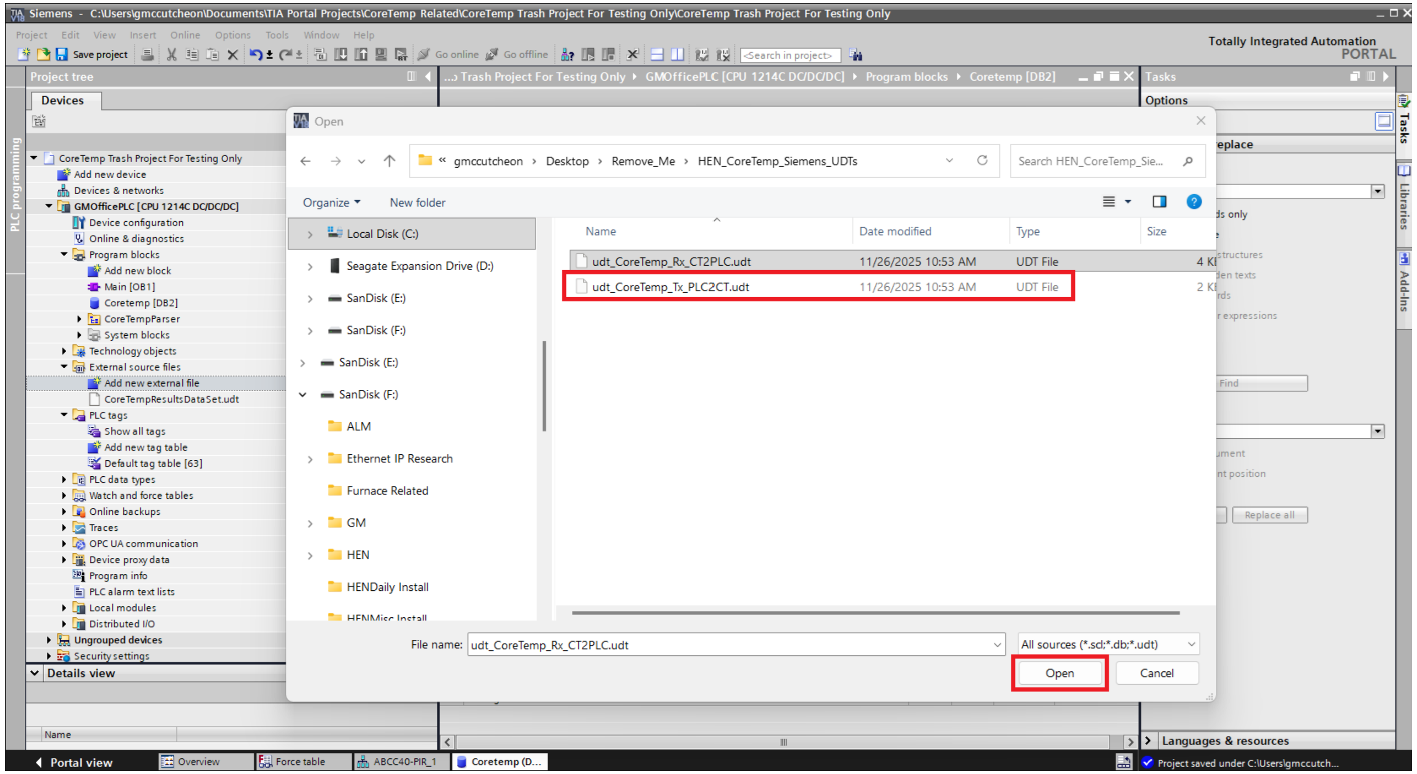
Add the udt_CoreTemp_Tx_PLC2CT.udt source files to the project

- Click External source files → Add new external source once more



Add the udt_CoreTemp_Tx_PLC2CT.udt source files to the project

- Browse to udt_CoreTemp_Tx_PLC2CT.udt
- Select it and click Open.



Siemens - C:\Users\gmccutcheon\Documents\TIA Portal Projects\CoreTemp Related\CoreTemp Trash Project For Testing Only\CoreTemp Trash Project For Testing Only

…› Trash Project For Testing Only › GMOfficePLC [CPU 1214C DC/DC/DC] › Program blocks › Coretemp [DB2]

« gmccutcheon › Desktop › Remove_Me › HEN_CoreTemp_Siemens_UDTs

Name	Date modified	Type	Size
udt_CoreTemp_Rx_CT2PLC.udt	11/26/2025 10:53 AM	UDT File	4 KB
udt_CoreTemp_Tx_PLC2CT.udt	11/26/2025 10:53 AM	UDT File	2 KB

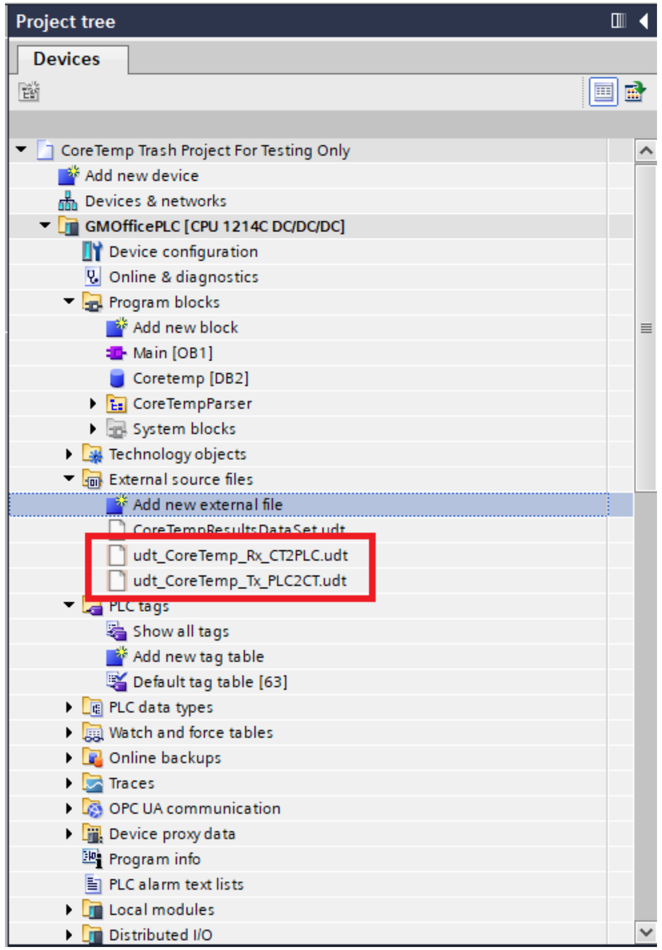
File name: udt_CoreTemp_Rx_CT2PLC.udt

All sources (*.scl;*.db;*.udt)

Project saved under C:\Users\gmccutch...

Verify .udt files in the project

- You should now see both files listed under External source files.



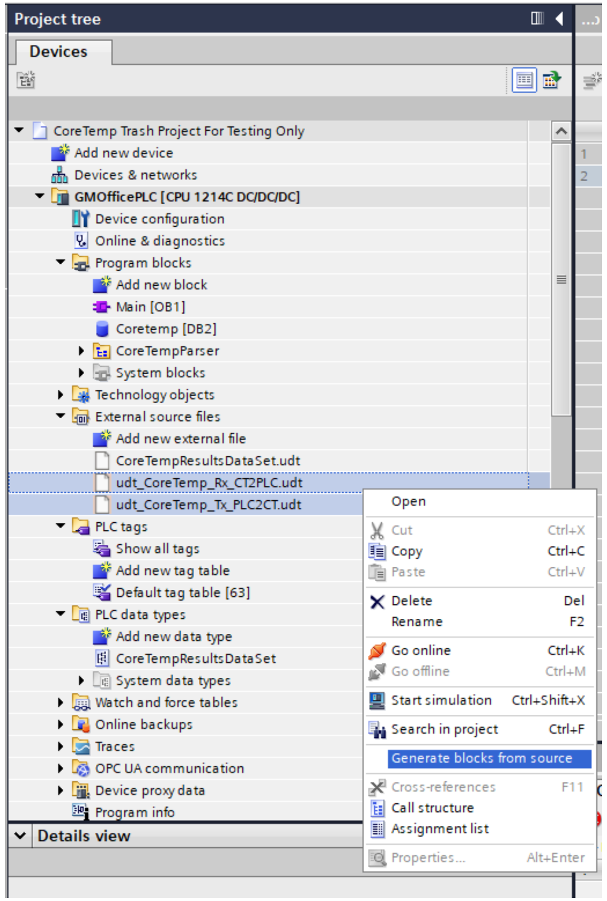
Project tree

Devices

- CoreTemp Trash Project For Testing Only
  - Add new device
  - Devices & networks
  - GMOfficePLC [CPU 1214C DC/DC/DC]
    - Device configuration
    - Online & diagnostics
    - Program blocks
      - Add new block
      - Main [OB1]
      - Coretemp [DB2]
      - CoreTempParser
      - System blocks
    - Technology objects
    - External source files
      - Add new external file
      - CoreTempResultsDataSet.udt
      - udt_CoreTemp_Rx_CT2PLC.udt
      - udt_CoreTemp_Tx_PLC2CT.udt
    - PLC tags
      - Show all tags
      - Add new tag table
      - Default tag table [63]
    - PLC data types
    - Watch and force tables
    - Online backups
    - Traces
    - OPC UA communication
    - Device proxy data
    - Program info
    - PLC alarm text lists
    - Local modules
    - Distributed I/O

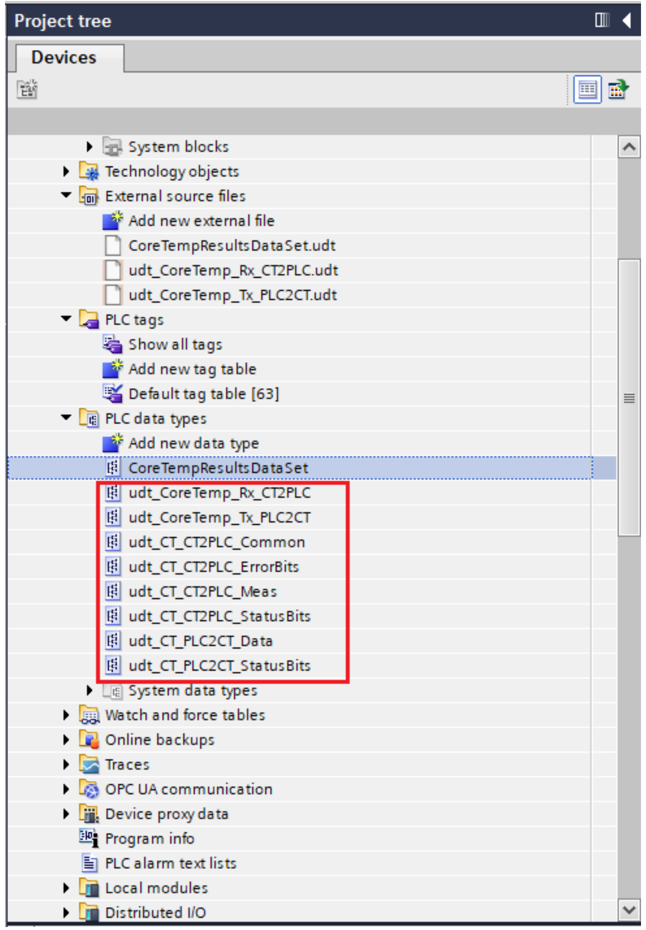
Generate the PLC data types from the sources

- Under External source files, select both CoreTemp .udt files (Ctrl-click).
- Right-click the selection
- Choose Generate blocks from source.
- TIA Portal parses each .udt file and creates the corresponding PLC data types under PLC data types for this CPU.



Verify the PLC data types were generated

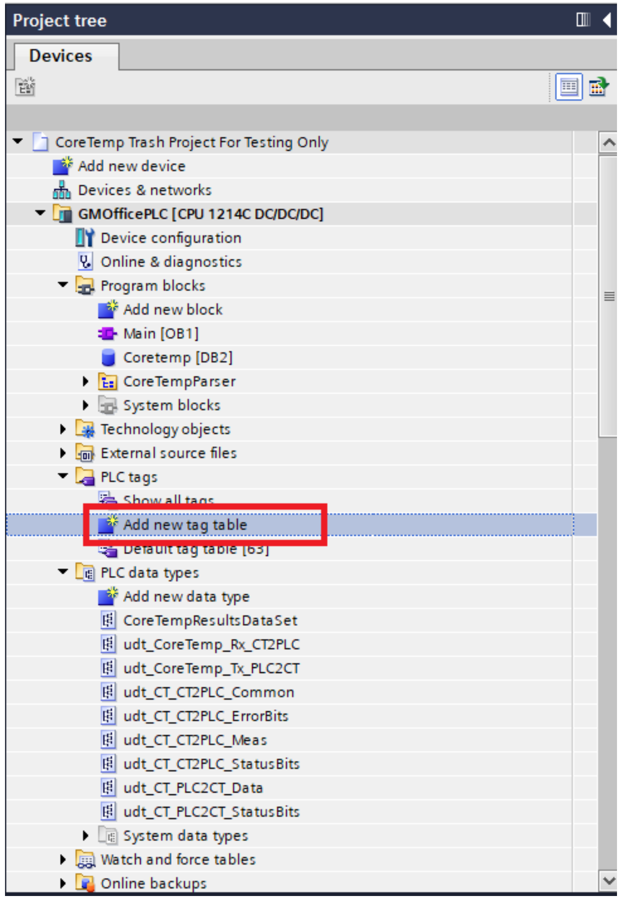
- Verify that the following types exist.
  - udt_CoreTemp_Rx_CT2PLC
  - udt_CT_CT2PLC_StatusBits
  - udt_CT_CT2PLC_ErrorBits
  - udt_CT_CT2PLC_Common
  - udt_CT_CT2PLC_Meas
- And
  - udt_CoreTemp_Tx_PLC2CT
  - udt_CT_PLC2CT_StatusBits
  - udt_CT_PLC2CT_Data



- System blocks
- Technology objects
- External source files
  - Add new external file
  - CoreTempResultsDataSet.udt
  - udt_CoreTemp_Rx_CT2PLC.udt
  - udt_CoreTemp_Tx_PLC2CT.udt
- PLC tags
  - Show all tags
  - Add new tag table
  - Default tag table [63]
- PLC data types
  - Add new data type
  - CoreTempResultsDataSet
  - udt_CoreTemp_Rx_CT2PLC
  - udt_CoreTemp_Tx_PLC2CT
  - udt_CT_CT2PLC_Common
  - udt_CT_CT2PLC_ErrorBits
  - udt_CT_CT2PLC_Meas
  - udt_CT_CT2PLC_StatusBits
  - udt_CT_PLC2CT_Data
  - udt_CT_PLC2CT_StatusBits
  - System data types
- Watch and force tables
- Online backups
- Traces
- OPC UA communication
- Device proxy data
- Program info
- PLC alarm text lists
- Local modules
- Distributed I/O

Create a new PLC tag table

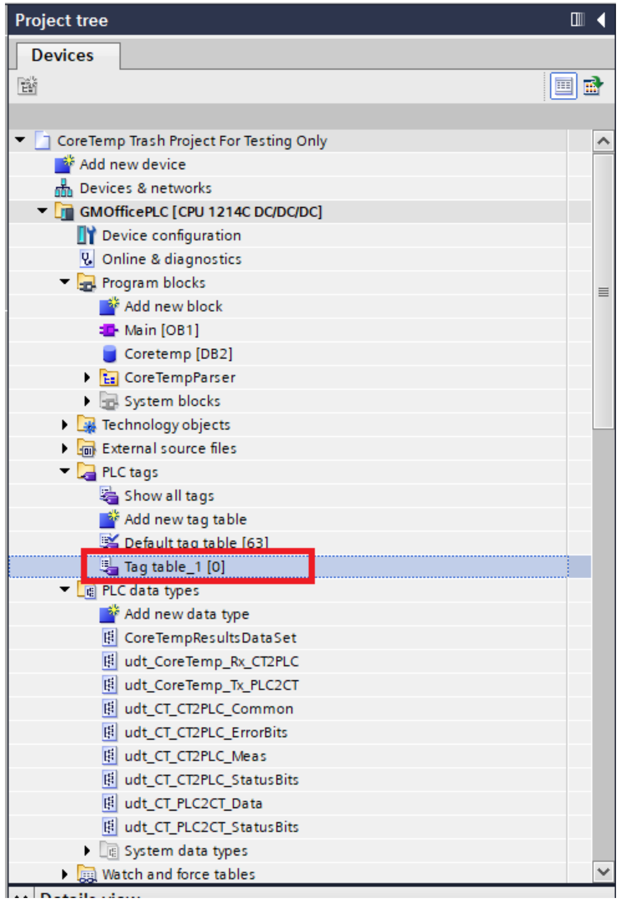
- In the project tree, expand PLC tags.
- Expand PLC Tags under the CPU.
- Click PLC tags→ Add new tag table
- A new empty tag table will be created
  - In this case named Tag table_1
  - (0) shows there are no tags in this table



Project tree

Devices

CoreTemp Trash Project For Testing Only
Add new device
Devices & networks
GMOfficePLC [CPU 1214C DC/DC/DC]
Device configuration
Online & diagnostics
Program blocks
Add new block
Main [OB1]
Coretemp [DB2]
CoreTempParser
System blocks
Technology objects
External source files
PLC tags
Show all tags
Add new tag table
Default tag table [63]
PLC data types
Add new data type
CoreTempResultsDataSet
udt_CoreTemp_Rx_CT2PLC
udt_CoreTemp_Tx_PLC2CT
udt_CT_CT2PLC_Common
udt_CT_CT2PLC_ErrorBits
udt_CT_CT2PLC_Meas
udt_CT_CT2PLC_StatusBits
udt_CT_PLC2CT_Data
udt_CT_PLC2CT_StatusBits
System data types
Watch and force tables
Online backups



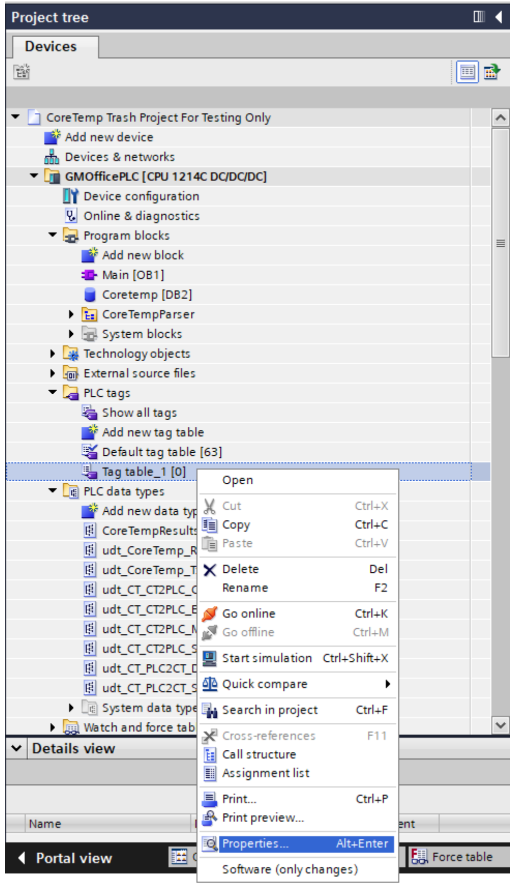
Project tree

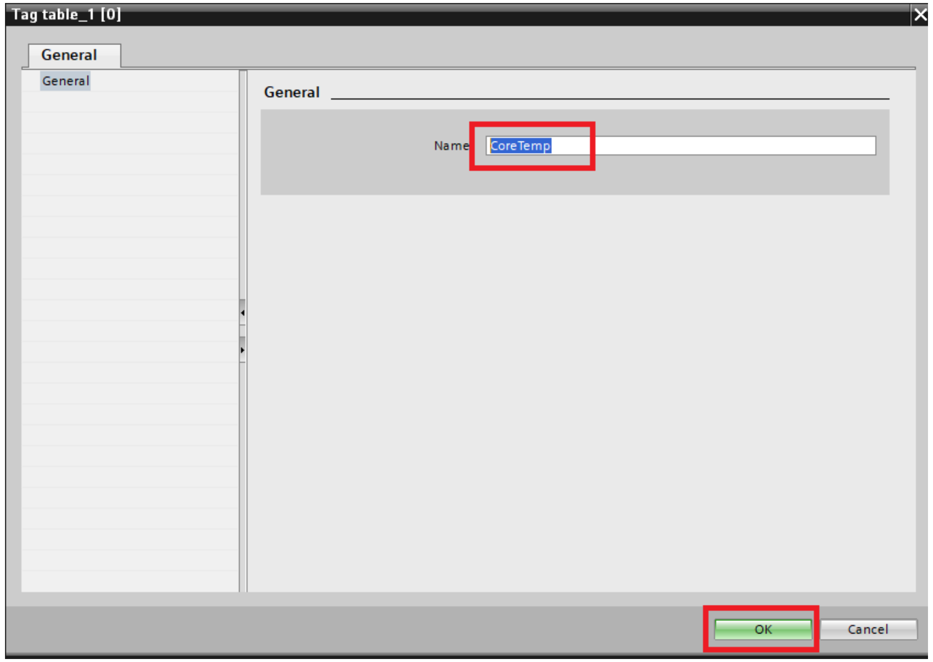
Devices

CoreTemp Trash Project For Testing Only
Add new device
Devices & networks
GMOfficePLC [CPU 1214C DC/DC/DC]
Device configuration
Online & diagnostics
Program blocks
Add new block
Main [OB1]
Coretemp [DB2]
CoreTempParser
System blocks
Technology objects
External source files
PLC tags
Show all tags
Add new tag table
Default tag table [63]
Tag table_1 [0]
PLC data types
Add new data type
CoreTempResultsDataSet
udt_CoreTemp_Rx_CT2PLC
udt_CoreTemp_Tx_PLC2CT
udt_CT_CT2PLC_Common
udt_CT_CT2PLC_ErrorBits
udt_CT_CT2PLC_Meas
udt_CT_CT2PLC_StatusBits
udt_CT_PLC2CT_Data
udt_CT_PLC2CT_StatusBits
System data types
Watch and force tables

Name the CoreTemp PLC tag table

- Select the new PLC tag table.
- Right-click the selection
- Choose Properties.
- Change the name to ‘CoreTemp’.
- Click OK.

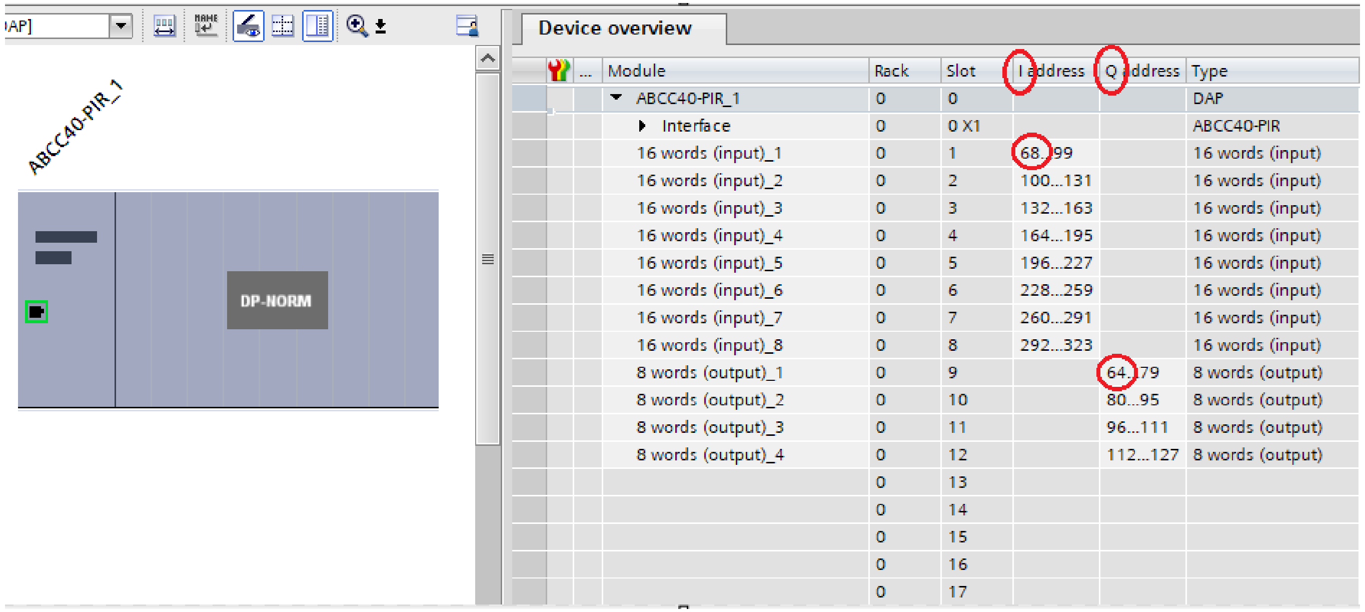




Name CoreTemp

Note the PLC start addresses

- The %IB68 / %QB64 example here assumes the CoreTemp interface is configured so that:
  - its first input byte is at %I68
  - its first output byte is at %Q64
- Use the actual start addresses from your own device configuration.

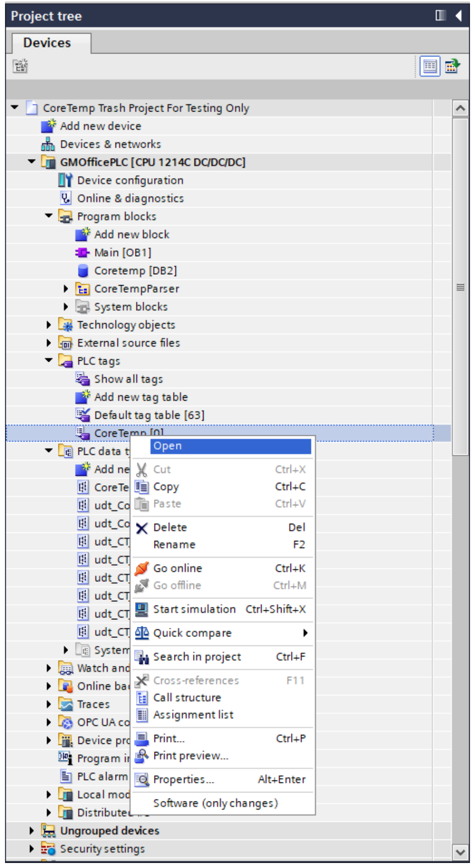


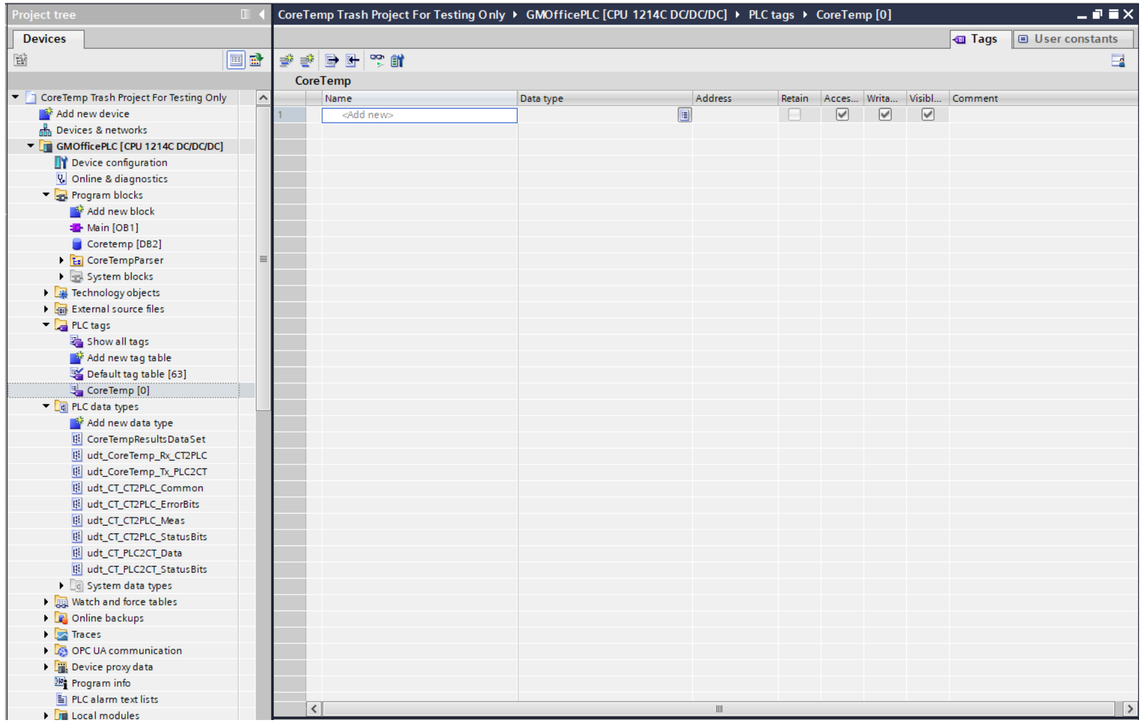
Device overview

Module	Rack	Slot	I address	Q address	Type
ABCC40-PIR_1	0	0			DAP
Interface	0	0 X1			ABCC40-PIR
16 words (input)_1	0	1	68...99		16 words (input)
16 words (input)_2	0	2	100...131		16 words (input)
16 words (input)_3	0	3	132...163		16 words (input)
16 words (input)_4	0	4	164...195		16 words (input)
16 words (input)_5	0	5	196...227		16 words (input)
16 words (input)_6	0	6	228...259		16 words (input)
16 words (input)_7	0	7	260...291		16 words (input)
16 words (input)_8	0	8	292...323		16 words (input)
8 words (output)_1	0	9		64...79	8 words (output)
8 words (output)_2	0	10		80...95	8 words (output)
8 words (output)_3	0	11		96...111	8 words (output)
8 words (output)_4	0	12		112...127	8 words (output)
	0	13			
	0	14			
	0	15			
	0	16			
	0	17			

Open the CoreTemp PLC tag table

- Select the new CoreTemp PLC tag table and double-click to open or
- Right-click the selection
- Choose Open.
- This will open the new CoreTemp tag table.



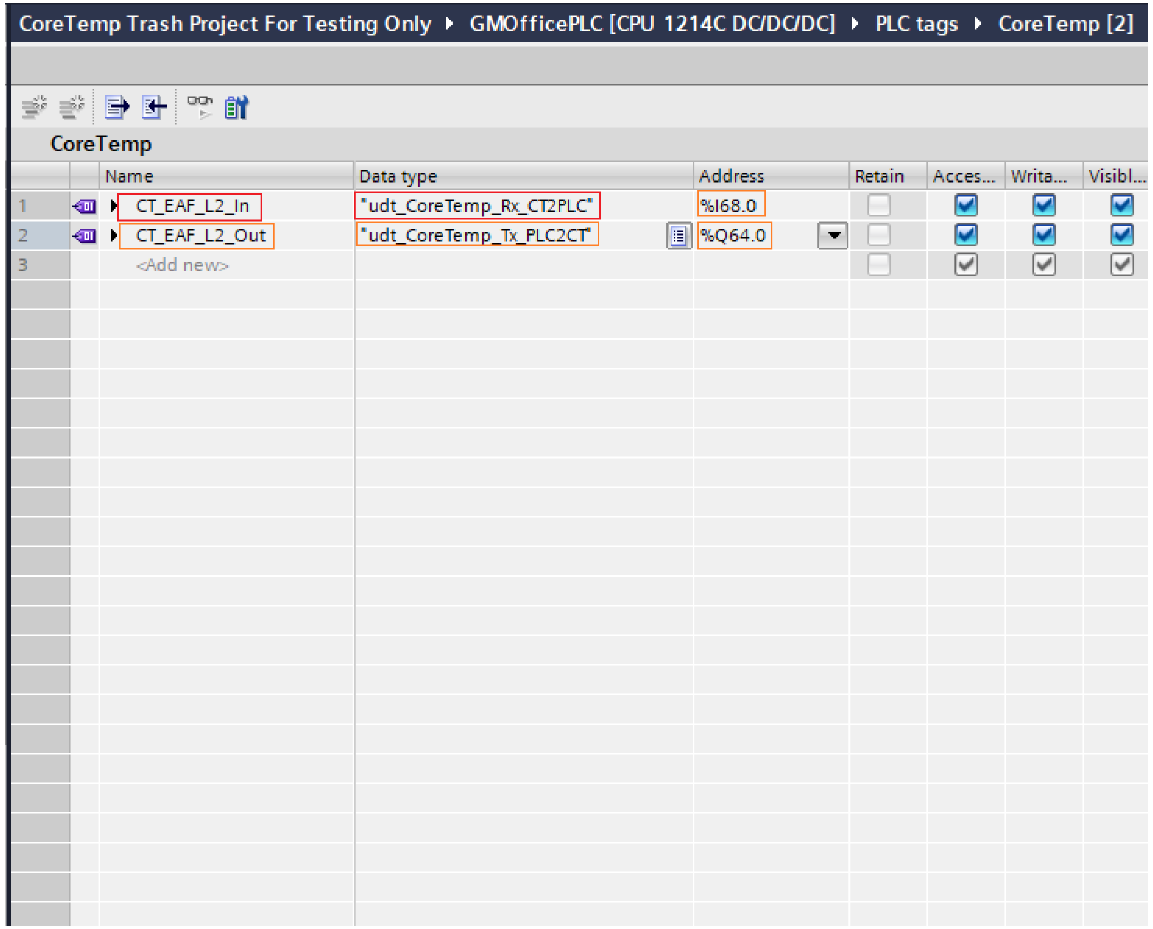


	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	<Add new>			□	☑	☑	☑	

Add tags to the CoreTemp PLC tag table

- Add CoreTemp → PLC tag:
  - Name = CT_EAF_L2_In
  - Data type = udt_CoreTemp_Rx_CT2PLC
  - Address = %I68.0
    - Note: references the starting input addresses' **first bit**.
- Add PLC → CoreTemp tag:
  - Name = CT_EAF_L2_Out
  - Data type = udt_CoreTemp_Tx_PLC2CT
  - Address = %Q64.0
    - Note: references the starting output addresses' **first bit**.

- **Important:** Use the actual start addresses from your own device configuration.



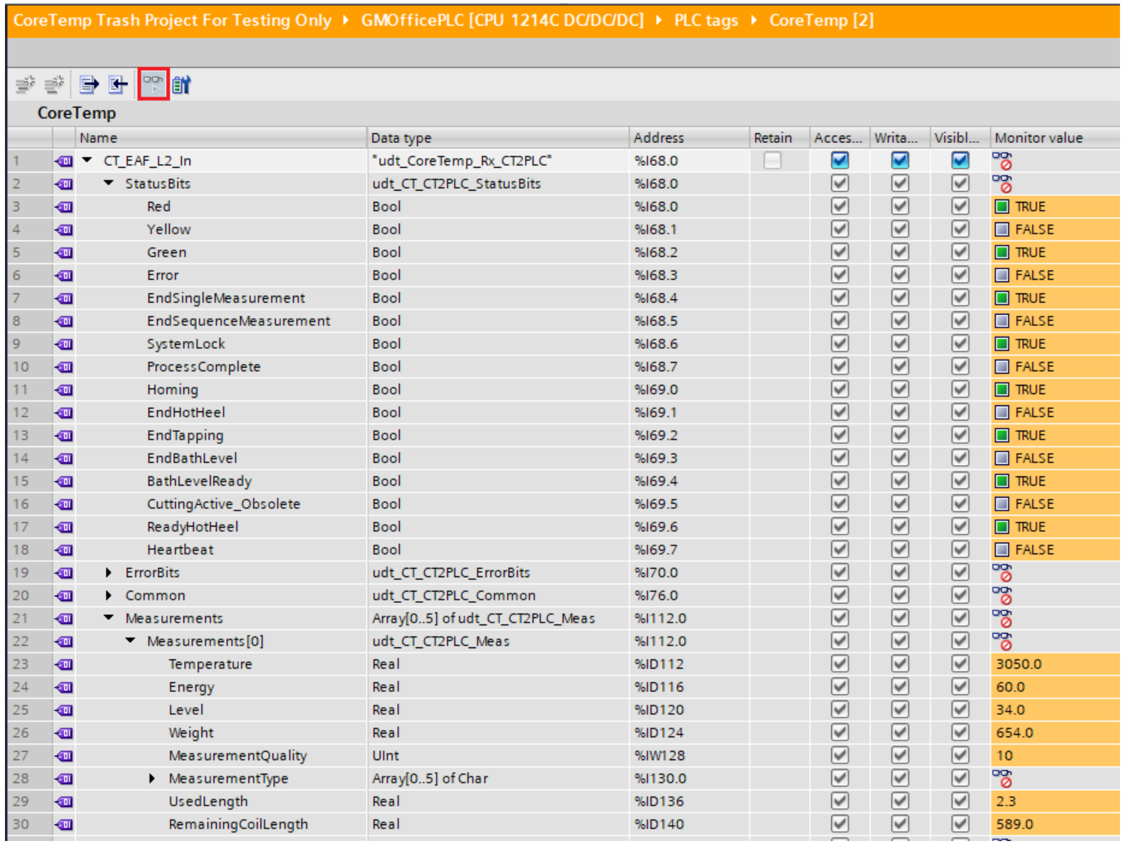
CoreTemp Trash Project For Testing Only ▸ GMOfficePLC [CPU 1214C DC/DC/DC] ▸ PLC tags ▸ CoreTemp [2]

CoreTemp

	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...
1	CT_EAF_L2_In	"udt_CoreTemp_Rx_CT2PLC"	%I68.0	□	☑	☑	☑
2	CT_EAF_L2_Out	"udt_CoreTemp_Tx_PLC2CT"	%Q64.0	□	☑	☑	☑
3	<Add new>			□	☑	☑	☑

Monitor values on the CoreTemp PLC tag table

- Compile the program.
- Upload to the PLC.
- Go online with the PLC.
- Open the CoreTemp tag table.
- Expand CT_EAF_L2_In → Status.
- Expand CT_EAF_L2_In Meas→ Meas[0].
- Click the Monitor All icon to read values from the PLC.



CoreTemp Trash Project For Testing Only ▸ GMOfficePLC [CPU 1214C DC/DC/DC] ▸ PLC tags ▸ CoreTemp [2]

CoreTemp

	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Monitor value
1	▼ CT_EAF_L2_In	"udt_CoreTemp_Rx_CT2PLC"	%I68.0	☐	☑	☑	☑	
2	▼ StatusBits	udt_CT_CT2PLC_StatusBits	%I68.0		☑	☑	☑	
3	Red	Bool	%I68.0		☑	☑	☑	TRUE
4	Yellow	Bool	%I68.1		☑	☑	☑	FALSE
5	Green	Bool	%I68.2		☑	☑	☑	TRUE
6	Error	Bool	%I68.3		☑	☑	☑	FALSE
7	EndSingleMeasurement	Bool	%I68.4		☑	☑	☑	TRUE
8	EndSequenceMeasurement	Bool	%I68.5		☑	☑	☑	FALSE
9	SystemLock	Bool	%I68.6		☑	☑	☑	TRUE
10	ProcessComplete	Bool	%I68.7		☑	☑	☑	FALSE
11	Homing	Bool	%I69.0		☑	☑	☑	TRUE
12	EndHotHeel	Bool	%I69.1		☑	☑	☑	FALSE
13	EndTapping	Bool	%I69.2		☑	☑	☑	TRUE
14	EndBathLevel	Bool	%I69.3		☑	☑	☑	FALSE
15	BathLevelReady	Bool	%I69.4		☑	☑	☑	TRUE
16	CuttingActive_Obsolete	Bool	%I69.5		☑	☑	☑	FALSE
17	ReadyHotHeel	Bool	%I69.6		☑	☑	☑	TRUE
18	Heartbeat	Bool	%I69.7		☑	☑	☑	FALSE
19	▶ ErrorBits	udt_CT_CT2PLC_ErrorBits	%I70.0		☑	☑	☑	
20	▶ Common	udt_CT_CT2PLC_Common	%I76.0		☑	☑	☑	
21	▼ Measurements	Array[0..5] of udt_CT_CT2PLC_Meas	%I112.0		☑	☑	☑	
22	▼ Measurements[0]	udt_CT_CT2PLC_Meas	%I112.0		☑	☑	☑	
23	Temperature	Real	%ID112		☑	☑	☑	3050.0
24	Energy	Real	%ID116		☑	☑	☑	60.0
25	Level	Real	%ID120		☑	☑	☑	34.0
26	Weight	Real	%ID124		☑	☑	☑	654.0
27	MeasurementQuality	UInt	%IW128		☑	☑	☑	10
28	▶ MeasurementType	Array[0..5] of Char	%I130.0		☑	☑	☑	
29	UsedLength	Real	%ID136		☑	☑	☑	2.3
30	RemainingCoilLength	Real	%ID140		☑	☑	☑	589.0

Using the tags in logic

- Once the tags are mapped, you can work directly with structured data.

```
// Example: read latest valid single measurement
IF CT_EAF_L2_In.Status.EndSingleMeas
   AND NOT CT_EAF_L2_In.Error.ColdMeasurement
   AND NOT CT_EAF_L2_In.Error.MajorScrap THEN

    LatestMeasurementTemp  := CT_EAF_L2_In.Meas[0].Temperature;
    LatestMeasurementMWh   := CT_EAF_L2_In.Meas[0].Energy;
    LatestMeasurementLevel := CT_EAF_L2_In.Meas[0].Level;
END_IF;

// Example: command a new single measurement
CT_EAF_L2_Out.Status.SystemReady     := TRUE;   // Interlocks ok
CT_EAF_L2_Out.Status.StartSingleMeas := TRUE;   // Command CoreTemp
```