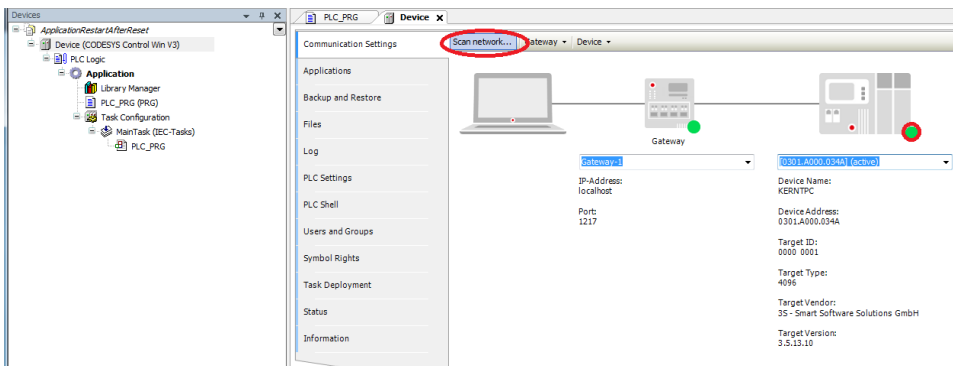
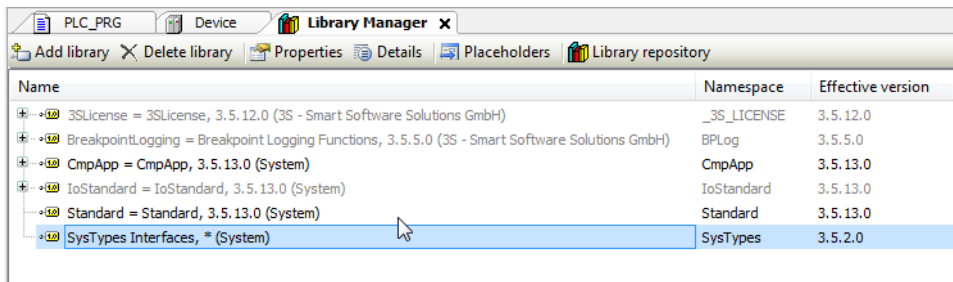


Automatic Restart of the Application after Resetting from IEC Code

- Create a "Standard project" and select *CODESYS Control Win V3* as the device.
- Define the target system by means of the *Network scan*.



- Open the *Library Manager* and add the following libraries:
CmpApp
SysTypes2 interfaces



- Create a persistent variable named *PersistentVars*.

Adapt *PersistentVars* as follows:

Declaration

```
{attribute 'qualified_only'}
VAR_GLOBAL PERSISTENT RETAIN
    xAutoRestart    : BOOL;
    udiCnt          : UDINT;
END_VAR
```

- Adapt the POU *PLC_PRG* as follows:

Declaration

```

VAR
    sAppName      :   STRING := 'Application';
    result         :   RTS_IEC_RESULT;

    pApp           :   POINTER TO CmpApp.APPLICATION;
    udiState       :   UDINT;
    udiOpState     :   UDINT;
    xInit          :   BOOL := TRUE;
    xReset         :   BOOL;
END_VAR

```

Implemen
tation

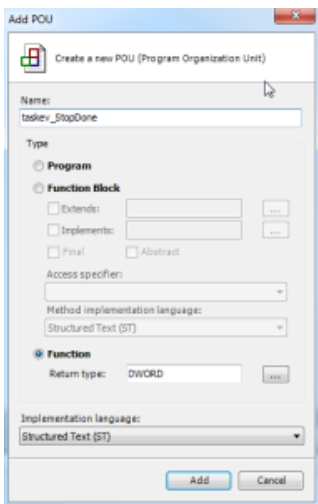
```

IF xInit THEN
    pApp := CmpApp.AppFindApplicationByName(pszString := sAppName, pResult := ADR(result));
    xInit := False;
    PersistentVars.xAutoRestart := False;
END_IF

IF xReset THEN
    xReset := FALSE;
    PersistentVars.xAutoRestart := True;
    CmpApp.AppReset(pApp := pApp, usResetOption := CmpApp.RTS_RESET);
END_IF

```

- Create a new function named *taskew_StopDone* with the return type *DWORD*.



Then adapt the function as follows:

Declaration

```

FUNCTION taskew_StopDone : DWORD
VAR_INPUT
    EventPrm: CmpApp.EVTPARAM_CmpAppStop;
END_VAR
VAR
END_VAR

```

```
IF PersistentVars.xAutoRestart THEN
    PersistentVars.udiCnt := PersistentVars.udiCnt + 1;
    CmpApp.AppStartApplication(PLC_PRG.pApp);
END_IF
```

-
- The screenshot shows the SIMATIC Manager interface. On the left, the 'Devices' tree is expanded to 'PLC Logic' under 'Application'. The 'Task Configuration' node is highlighted. On the right, the 'System Events' tab is active, showing a table with columns 'Name' and 'Description'. The 'Add Event Handler' button is highlighted with a red circle.

Add Event Handler

Event: **StopDone**

Function to call: **taskv_StopDone**

Scope: ☒ Application ☐ POUs

Implementation language: **Structured Text (ST)**

Description: Called after application stops. Context=Communicationtask. Debugging=Disabled

OK Cancel

Task Groups Monitor Variable Usage System Events Properties						
Add Event Handler Remove Event Handler Event Info... Open Event Function						
Name	Description	Context	Debugging	Function to call	Active	
StopDone	Called after application stops	Communication task		taskev_StopDone	☒	

-
- The screenshot shows the Siemens SIMATIC Manager interface. On the left, the 'Devices' tree is expanded, showing the project structure. The 'Application [run]' window is active, displaying the 'Device.Application.PersistantVars' table. The table has columns for Expression, Type, Value, Prepared value, Address, and Comment. The 'Value' column shows 'FALSE' for 'xAutoRestart' and '5' for 'udiCnt'.
- | Expression | Type | Value | Prepared value | Address | Comment |
|--------------|-------|-------|----------------|---------|---------|
| xAutoRestart | BOOL | FALSE | | | |
| udiCnt | UDINT | 5 | | | |



The variable `PersistentVars.udlCnt` is incremented **from the application** with each restart.