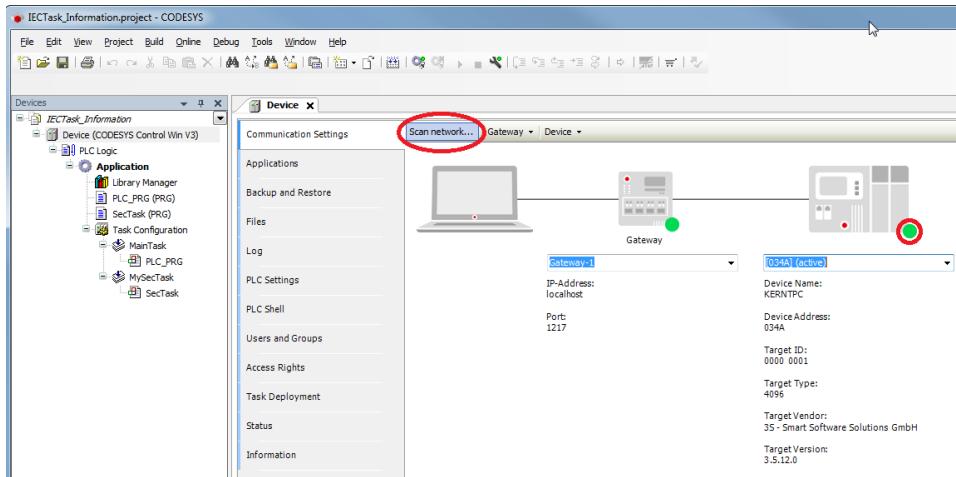
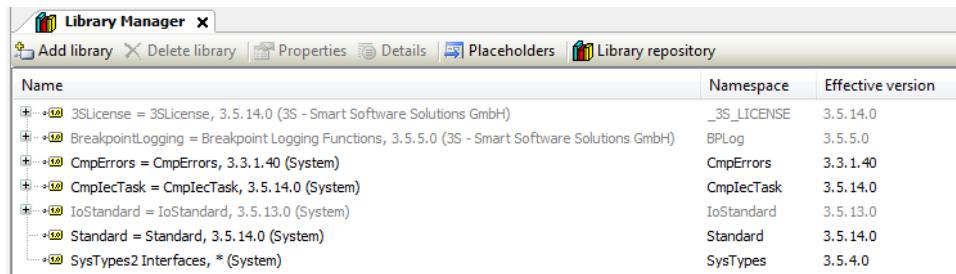


Task Configuration: Reading the Cycle Time and Other Information

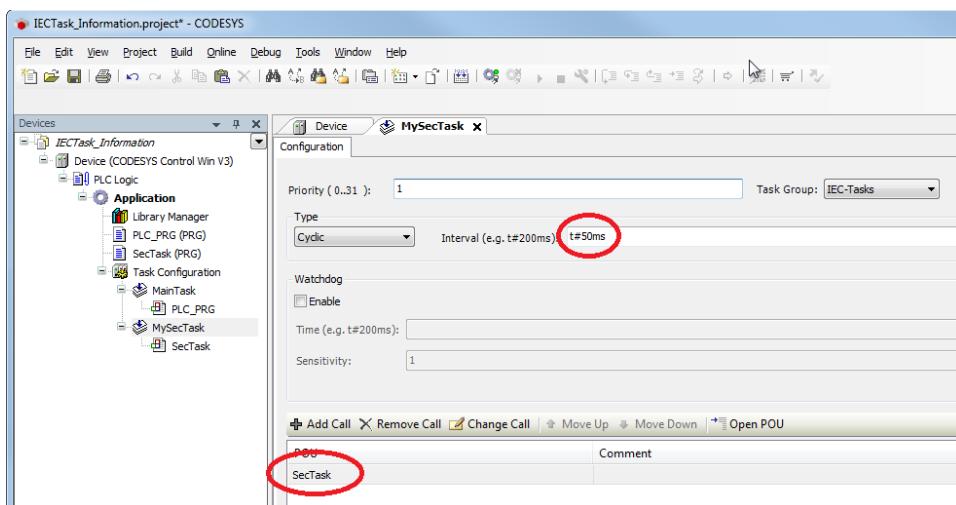
- Create a "Standard project" and select *CODESYS Control Win V3* as the device.
- Define the target system by means of the *Network scan*.



- Open the *Library Manager* and add the following libraries:
CompErrors
CmpIecTask
SysTypes2 Interfaces



- Create a new POU named *sc* and a new task named *MySecTask*.
Assign the POU *SecTask* to the task *MySecTask* and set the call interval to 50 milliseconds.



- Adapt the POU *SecTask* as follows:

Declaration

```

VAR
    iIndex      :      INT;
    sIndex      :      STRING;
END_VAR

```

Implementation

```

FOR iIndex := 0 TO 1000 DO
    //Do something to raise the cycle time
    sIndex := INT_TO_STRING(iIndex);
END_FOR

```

- Adapt the POU [PLC_PRG](#) as follows:

Declaration

```

VAR
    dwCycleTimeMainTask, dwCycleTimeSecTask      :      DWORD;
    dwIntervalMainTask, dwIntervalSecTask         :      DWORD;
    sNameMainTask, sNameSecTask                  :      STRING;
    hFirstTask, hSecTask                        :      RTS_IEC_HANDLE;
    iecResult                                :      RTS_IEC_RESULT;
    sAppName                                  :      STRING := 'Application';
    pIecInfo                                 :      POINTER TO CmpIecTask.Task_Info2;
END_VAR

```

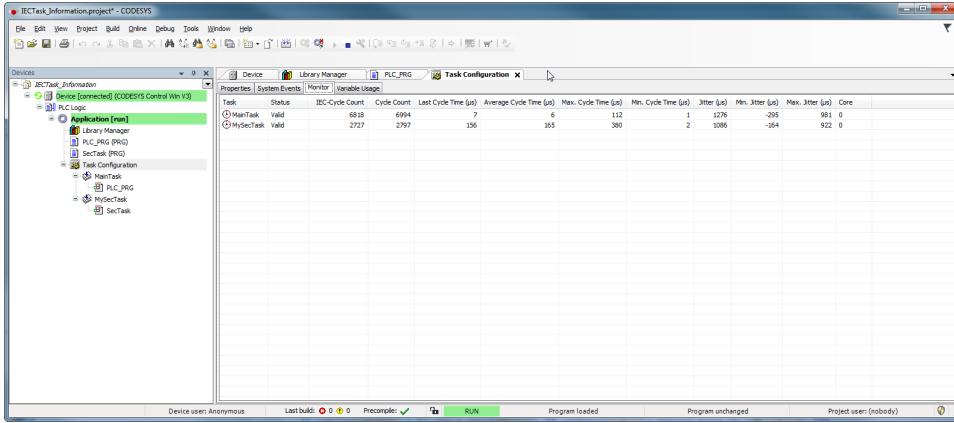
Implementation

```

hFirstTask := CmpIecTask.IecTaskGetFirst(pszAppName := ADR(sAppName), pResult := ADR(iecResult));
IF ((hFirstTask <> RTS_INVALID_HANDLE) AND (iecResult = CmpErrors.Errors.ERR_OK)) THEN
    pIecInfo := CmpIecTask.IecTaskGetInfo3(hIecTask := hFirstTask, pResult := ADR(iecResult));
    dwIntervalMainTask := pIecInfo^.dwInterval;
    dwCycleTimeMainTask := pIecInfo^.dwCycleTime;
    sNameMainTask := pIecInfo^.pszName^;
    hSecTask := IecTaskGetNext(pszAppName := ADR(sAppName), hPrevIecTask := hFirstTask, pResult :=
= ADR(iecResult));
    IF ((hSecTask <> RTS_INVALID_HANDLE) AND (iecResult = CmpErrors.Errors.ERR_OK)) THEN
        pIecInfo := CmpIecTask.IecTaskGetInfo3(hIecTask := hSecTask, pResult := ADR(iecResult));
        dwIntervalSecTask := pIecInfo^.dwInterval;
        dwCycleTimeSecTask := pIecInfo^.dwCycleTime;
        sNameSecTask := pIecInfo^.pszName^;
    END_IF
END_IF

```

- After you have uploaded and started the project on the controller, you can compare the values between the task configuration and the IEC code.



```

Device: IECTask_Information
File Edit View Project Build Online Debug Tools Window Help
Device Library Manager PLC_PRG Task Configuration
Properties System Events Monitor Variable Usage
Task Configuration
MainTask Valid 6994 7 6 127 1 -395 981 0
MySecTask Valid 2727 156 365 380 2 2088 -364 922 0

Device user: Anonymous Last build: 0 0 Precompile: ✓ RUN Program loaded Program unchanged Project user: (nobody)

Device: IECTask_Information
File Edit View Project Build Online Debug Tools Window Help
Device Library Manager PLC_PRG Task Configuration
Properties System Events Monitor Variable Usage
Task Configuration
Device: Application PLC_PRG
Variables
Type Value Prepared Value Address Comment
hCycleTimeMainTask DWORD 154
hCycleTimeSecTask DWORD 5
hMainTask DWORD 50000
hMainTaskTask STRING MySecTask
hMainTaskTaskTask STRING MySecTask
hMainTaskTaskTaskTask STRING MySecTask
hMainTaskTaskTaskTaskTaskTask PTR TO BYTE 16#00028C
hMainTaskTaskTaskTaskTaskTaskTask PTR TO BYTE 16#00027C
hResult UDINT 0
sAppName STRING Application
sTaskInfo PTR TO Cingle 16#00A098
1.0 IF (hFirstTask[16#00028C] != CmpIecTask.IecTaskGetFirst(pszAppName := ADDR(appName, AppName), pResult := ADDR(iecResult, 0))) THEN
2.0   IF (hFirstTask[16#00028C] > RT5_INVALID_HANDLE[16#FFFF]) AND (iecResult[0] = CmpErrors.Errors_Error_Ok[0]) THEN
3.0     dwIntervalMainTask := 50000 := pIecInfo->dwInterval[16#00027C]
4.0     dwIntervalMainTask := 50000 := pIecInfo->dwInterval[16#00028C]
5.0     dwCycleTimeMainTask := 154 := pIecInfo->dwCycleTime[16#00028C]
6.0     dwCycleTimeMainTask := 154 := pIecInfo->dwCycleTime[16#00027C]
7.0     hMainTask := hFirstTask := pIecTask := hFirstTask[16#00028C] := CmpIecTask.IecTaskGetNext(pszAppName := ADDR(appName, AppName), hPrevIecTask := hFirstTask[16#00028C], pResult := ADDR(iecResult, 0))
8.0     IF (hFirstTask[16#00028C] > RT5_INVALID_HANDLE[16#FFFF] AND (iecResult[0] = CmpErrors.Errors_Error_Ok[0])) THEN
9.0       dwCycleTimeSecTask := 5 := pIecInfo->dwCycleTime[16#00028C]
10.0      dwCycleTimeSecTask := 5 := pIecInfo->dwCycleTime[16#00027C]
11.0      dwCycleTimeSecTask := 5 := pIecInfo->dwCycleTime[16#00028C]
12.0      dwCycleTimeSecTask := 5 := pIecInfo->dwCycleTime[16#00027C]
13.0      dwMainTask := hMainTask := pIecInfo->dwMainTask[16#00028C]
14.0    END_IF
15.0  END_IF
16.0 END_IF

```