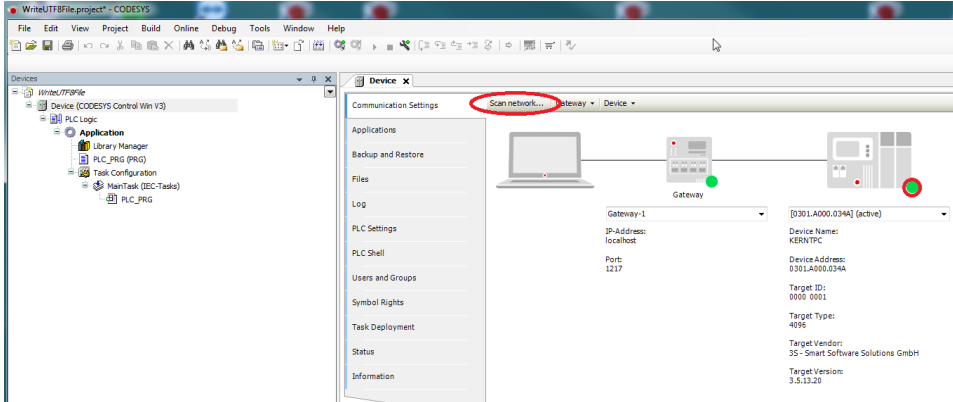


SysFile: Saving Files in UTF-8 Format

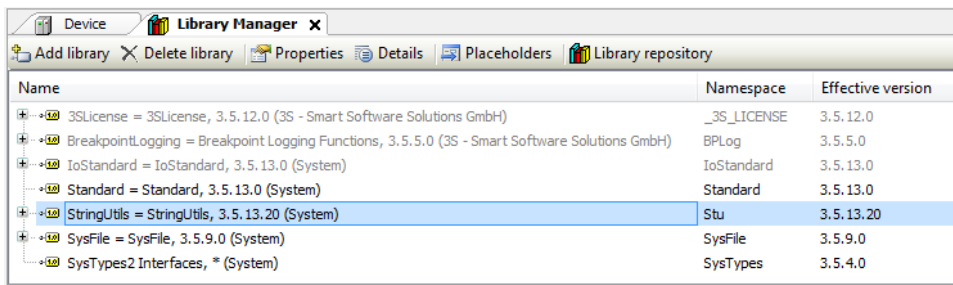
Data can be stored in UTF-8 format when working with the *WSTRING* type in the IEC code.

By using the ConvertUTF16toUTF8 function from the *StringUtils* library, the text is converted to UTF-8 format and can be saved to a file.

- 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:
StringUtils
SysFile
SysTypes2 Interfaces



- Edit the *PLC_PRG* POU as follows:

Declaration

```
VAR
    xWrite          : BOOL;
    szFileName      : STRING(128) := 'c:/tmp/TestUTF8.txt';
    hFile           : SysTypes.RTS_IEC_HANDLE := SysTypes.RTS_INVALID_HANDLE;
    iecResult       : SysTypes.RTS_IEC_RESULT;
    wsTextLine      : WSTRING(10) := "Übersetzer";
    sUTF8TextLine   : STRING;
    udiPos          : __XWORD;
    udiWrite        : __XWORD;
END_VAR
```

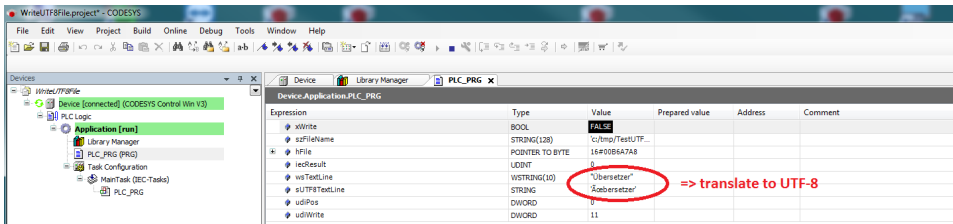
Implement ation

```

IF xWrite THEN
    xWrite := FALSE;
    hFile := SysFileOpen(szFile:=szFileName, am:=SysFile.AM_Write , pResult:=ADR(iecResult));
    IF hFile <> RTS_INVALID_HANDLE THEN
        Stu.ConvertUTF16toUTF8(sourceStart := ADR(wsTextLine), targetStart := ADR(sUTF8TextLine),
        dwTargetBufferSize := SIZEOF(wsTextLine), bStrictConversion := TRUE);
        udiWrite := SysFileWrite(hFile:=hFile, pbyBuffer:=ADR(sUTF8TextLine), ulSize:=INT_TO_UDINT(LEN
(sUTF8TextLine)), pResult:=ADR(iecResult));
        iecResult := SysFileClose(hFile:=hFile);
    END_IF
END_IF

```

- Start the project and set the *xWrite* variable to *TRUE*.



If the file is viewed in an editor, then the text is displayed correctly:

