

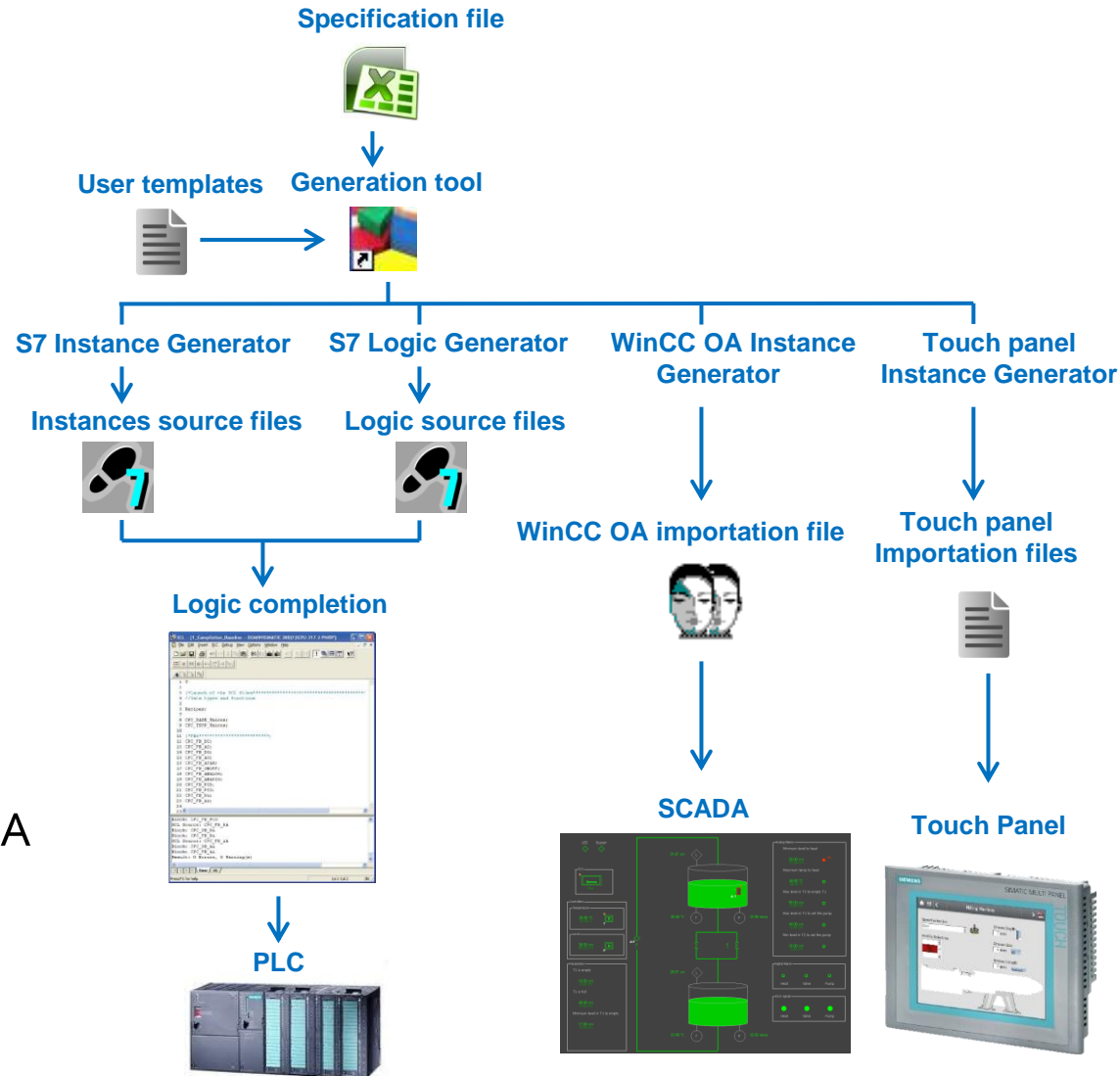


Deployment of the CPC6 control application

Marc Quilichini
marc.quilichini@cern.ch

CERN EN-ICE

1. Specification file
2. Generation
 - a) Instance
 - b) Logic
 - c) WinCC OA
 - d) Touch panel
3. Importation into S7
4. Logic completion
5. Downloading
6. WinCC OA project
7. Importation into WinCC OA
8. Design of panels for supervision



- ✓ Specification and generation tools: MS Office (Excel v2008), Java SE Runtime Environment (v6) and UAB (v1.6.x)
- ✓ PLC: Simatic Step 7 (v5.5)
- ✓ SCADA: PVSS(v3.8) / WinCC OA (v3.11)
- ✓ Touch panel: WinCC Flexible (v 2008 SP3) / TIAPortal (v12/v13)

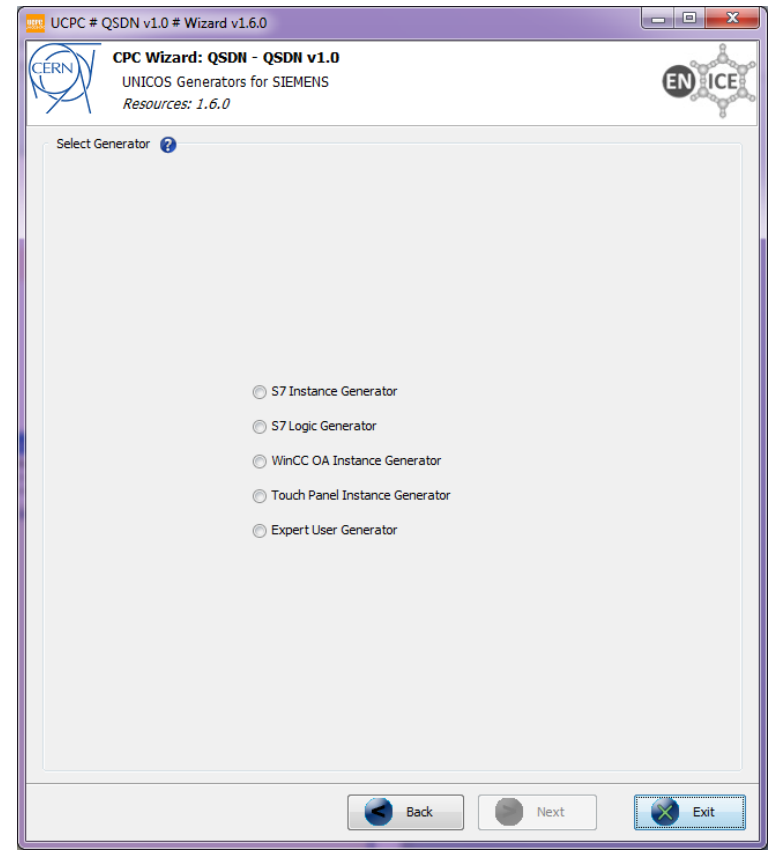


Creation of a Siemens S7 UNICOS-CPC 6 application

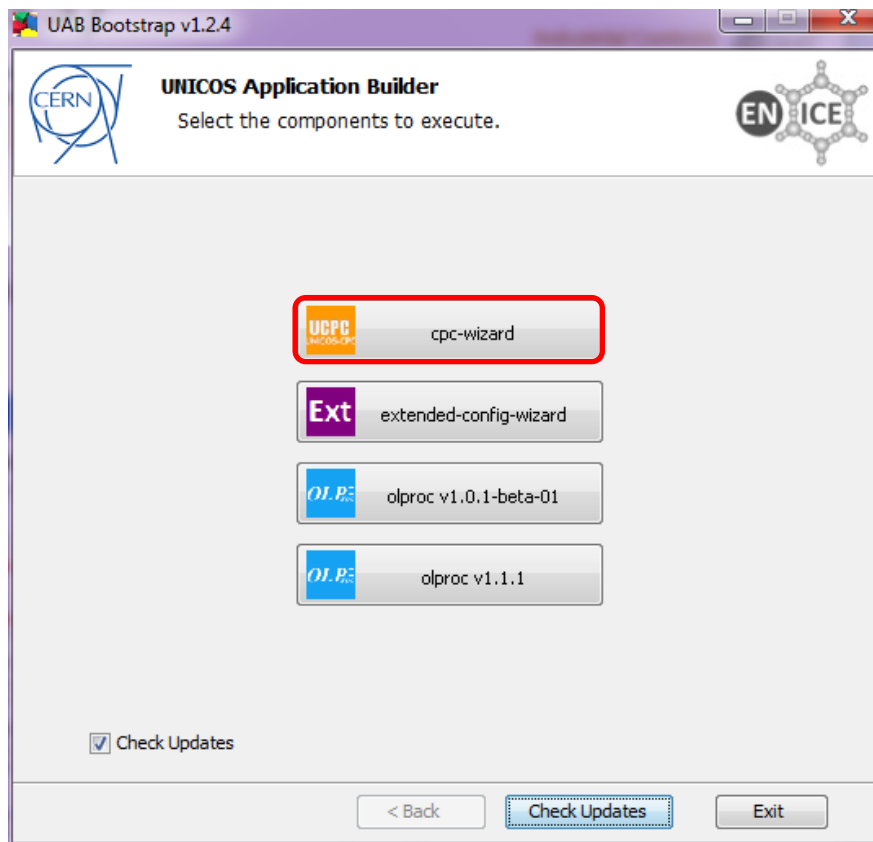
UAB Generation Tool

CPC-Wizard

- S7 Instance Generator
- S7 Logic Generator
- WinCC OA Instance Generator
- WinCC Flexible Instance Generator



1. Run the CPC-Wizard from the UAB Bootstrap by clicking on cpc-wizard.

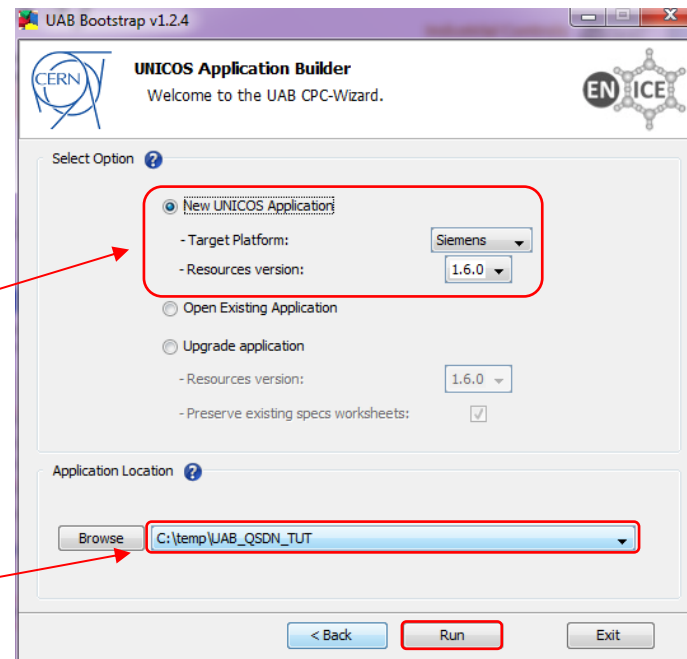


2. Create a new UNICOS application for Siemens platform

Choose Siemens platform and last version of the resource package

Choose a path for the application folder:

C:\UAB_QSDN



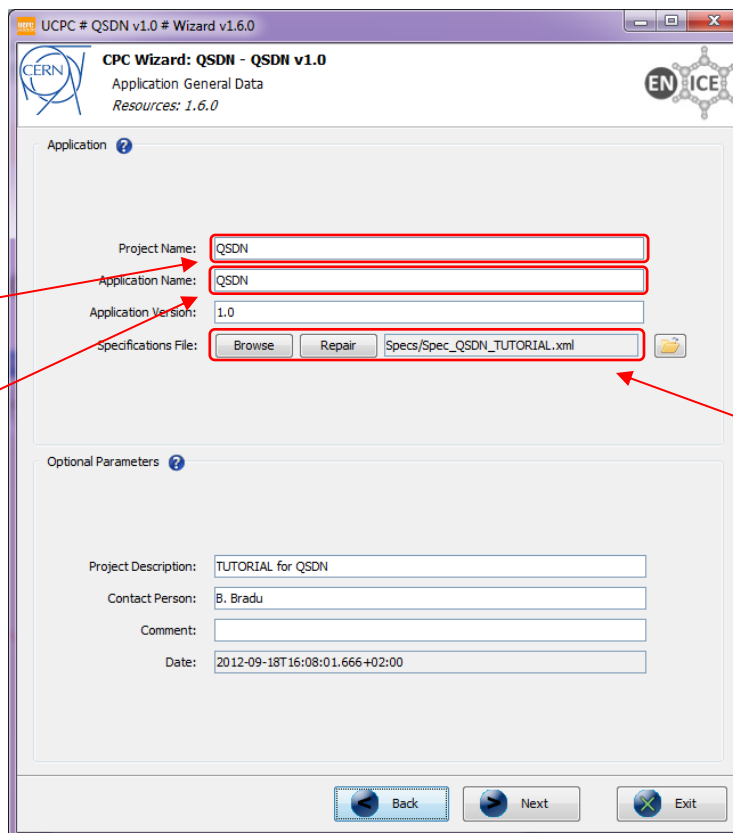
Click on Run

3. Fill in the "Application General Data" window

Write a project name:
UCPC6_course

Write an application name:
QSDN

Select your specification
file



UCPC # QSDN v1.0 # Wizard v1.6.0

CPC Wizard: QSDN - QSDN v1.0
Application General Data
Resources: 1.6.0

Application

Project Name: QSDN

Application Name: QSDN

Application Version: 1.0

Specifications File: Browse Repair Specs/Spec_QSDN_TUTORIAL.xml

Optional Parameters

Project Description: TUTORIAL for QSDN

Contact Person: B. Bradu

Comment:

Date: 2012-09-18T16:08:01.666+02:00

Back Next Exit

4. Fill in the "SIEMENS PLC Specifications" window according to your PLC configuration

Write a PLC name:
DEVICER17x

Write the corresponding IP address

Choose the corresponding PLC type

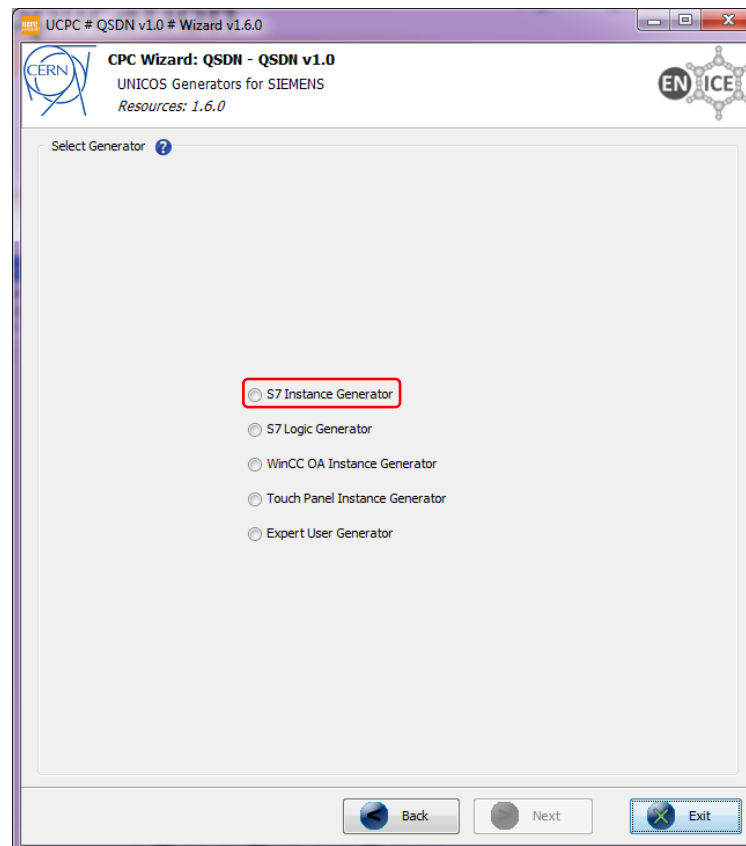
Set WinCC OA configuration

Set PLC connection configuration

The screenshot shows the 'CPC Wizard: QSDN - QSDN v1.0' window with the following configuration details:

- General Data:** PLC Name: CFP_LABO_BEN, IP Address: 137.138.192.1, PLC Type: S7-300 PN/DP
- WinCC OA Configuration:** Partner Rack: 0, Partner Slot: 0, Partner Con. Resource: 10, Timeout: 5000
- PLC Connection:** Local Id: 1, Local Rack: 0, Local Slot: 2, Local Con. Resource: 10
- Recipes:** Enable recipes: checked, Max. number of recipe values: 1000, Activation Timeout (s): 100
- User Resources Configuration:** Dynamic and Static User Resources File fields with Browse and Clear buttons.
- General Configuration:** Diagnostic Channel: checked, Event Buffer Size: 1000
- Address Configuration:** 1st FB: 100, 1st FC: 100, 1st DB: 100, 1st UDT: 100

UNICOS Generators for Siemens



5. S7 Instance Generator

The screenshot shows the 'CPC Wizard: QSDN - QSDN v1.0' window, specifically the 'S7 Instance Generator' tab. The interface includes several sections and controls:

- General Data:** Fields for 'Type Template Folder' and 'Output Folder'. Checkboxes for 'Process Semantic Rules' and 'Post Process User Template'.
- Global Files To Generate:** Checkboxes for 'Communication', 'CompilationInstances', 'CompilationBaseline', and 'CompilationOB'. A 'Global files scope' dropdown is set to 'All types'.
- Generate Symbols File:** A 'Symbols' checkbox.
- IO Commissioning:** A 'Generate Commissioning File' checkbox.
- UNICOS Types To Generate:** A table with columns 'Device Type' and 'Instances'. The table is populated with various device types and their instance counts.
- Buttons:** 'Select All', 'Edit Specs.', and 'Reload Specs.' buttons are located below the table.
- Generation Status:** A progress bar at 100% and a green checkmark icon.
- Bottom Panel:** Tabs for 'Instance', 'Logic', and 'Expert'. A 'Generate' button is highlighted.

Annotations with arrows point to specific elements:

- Communication and TSPP files to generate:** Points to the 'Communication' checkbox.
- Baseline compilation file:** Points to the 'CompilationBaseline' checkbox.
- Symbols file to generate:** Points to the 'Symbols' checkbox.
- To select all object types:** Points to the 'Select All' button.
- Instance compilation file:** Points to the 'CompilationInstances' checkbox.
- Organization blocks compilation file to generate:** Points to the 'CompilationOB' checkbox.
- To generate the IO commissioning file:** Points to the 'Generate Commissioning File' checkbox.
- To open and edit the specification file:** Points to the 'Edit Specs.' button.
- To reload the specification file in the table:** Points to the 'Reload Specs.' button.
- It means the specification file has changed:** Points to the yellow warning triangle icon in the 'Generation Status' section.

Device Type	Instances
AnaDO	0
Analog	10
AnalogAlarm	22
AnalogDigital	3
AnalogInput	18
AnalogInputReal	9
AnalogOutput	8
AnalogOutputReal	9
AnalogParameter	144

6. S7 Logic Generator

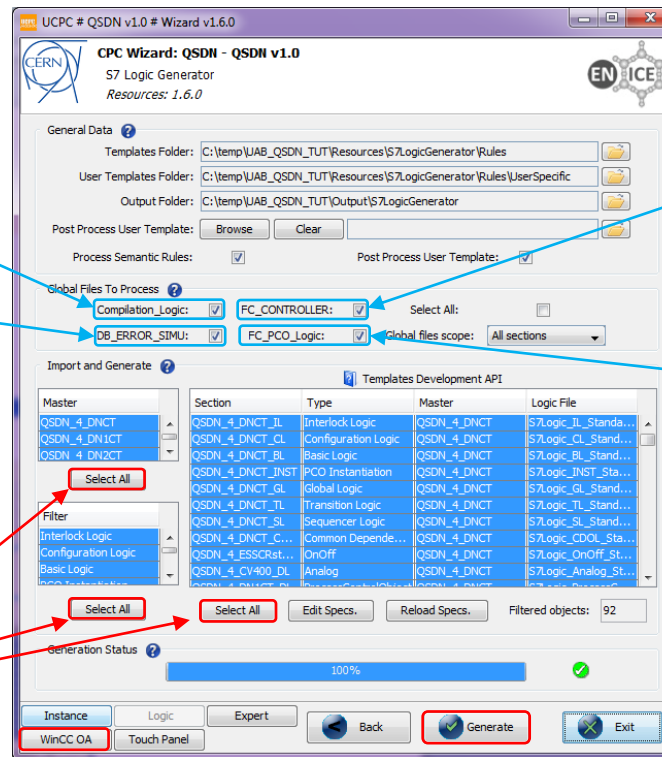
It generates the compilation file

It generates the files for Error/Simu conditions

It generates the file for the PID execution

It generates a function to call all logic functions

To generate all logic files

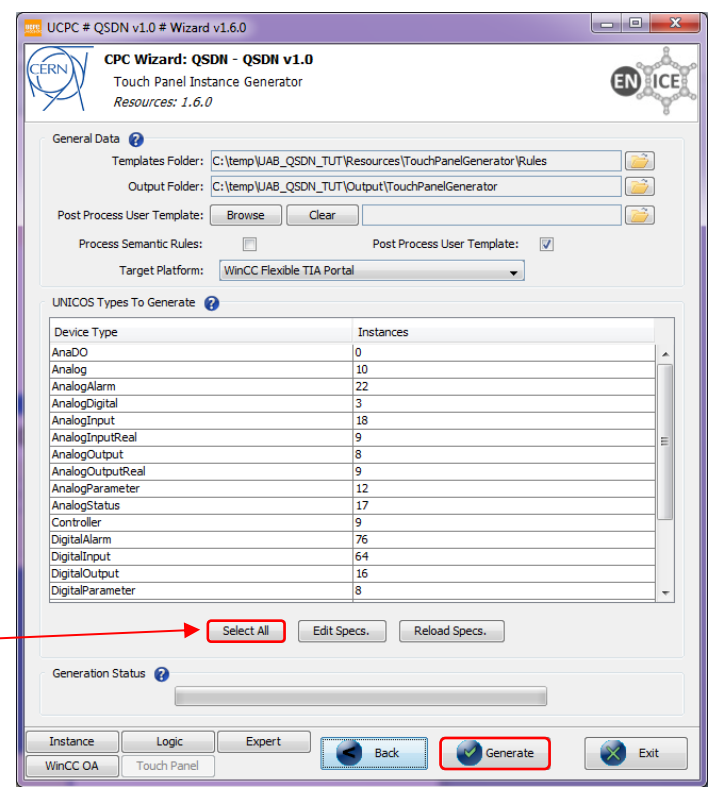


7. WinCC OA Instance Generator

The screenshot shows the 'CPC Wizard: QSDN - QSDN v1.0' window. The 'General Data' section includes fields for Templates Folder, Output Folder, DB File, and Post Process User Template. A blue arrow points to the 'Process Semantic Rules' checkbox, which is checked. Below this is a table of UNICOS Types To Generate with columns for Device Type and Instances. A red arrow points to the 'Select All' button below the table. At the bottom, the 'Touch Panel' button is highlighted with a red box, and the 'Generate' button is also highlighted with a red box.

Device Type	Instances
AnaDO	0
Analog	10
AnalogAlarm	22
AnalogDigital	3
AnalogInput	18
AnalogInputReal	9
AnalogOutput	8
AnalogOutputReal	9
AnalogParameter	12
AnalogStatus	17
Controller	9
DigitalAlarm	75
DigitalInput	64
DigitalOutput	16
DigitalParameter	8

8. Touch Panel Instance Generator



To select all the objects



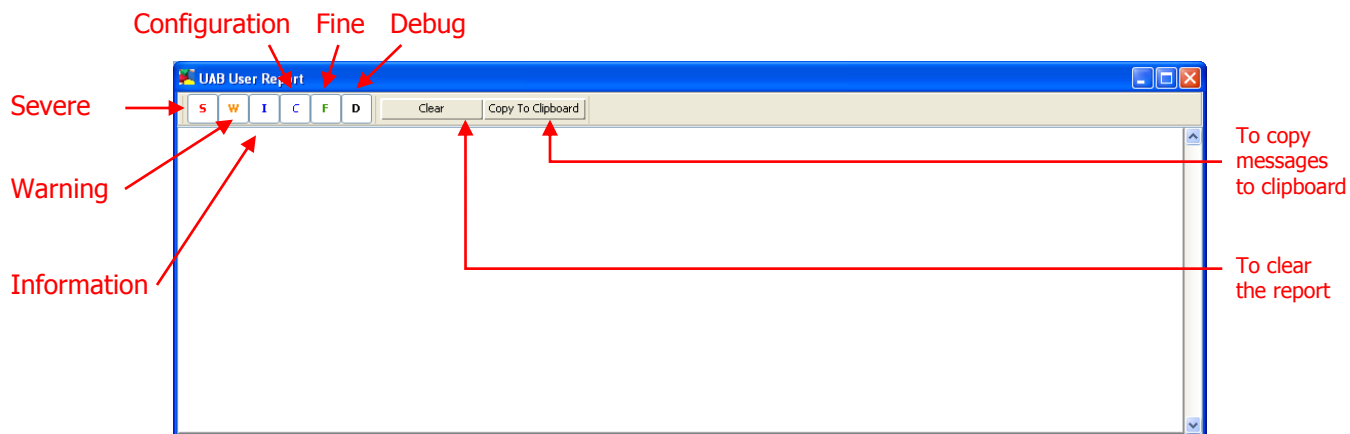
Creation of a Siemens S7 UNICOS-CPC 6 application

Generated files

UAB_project\Output

- S7InstanceGenerator
- S7LogicGenerator
- TouchPanelGenerator
- WinCCOAIInstanceGenerator

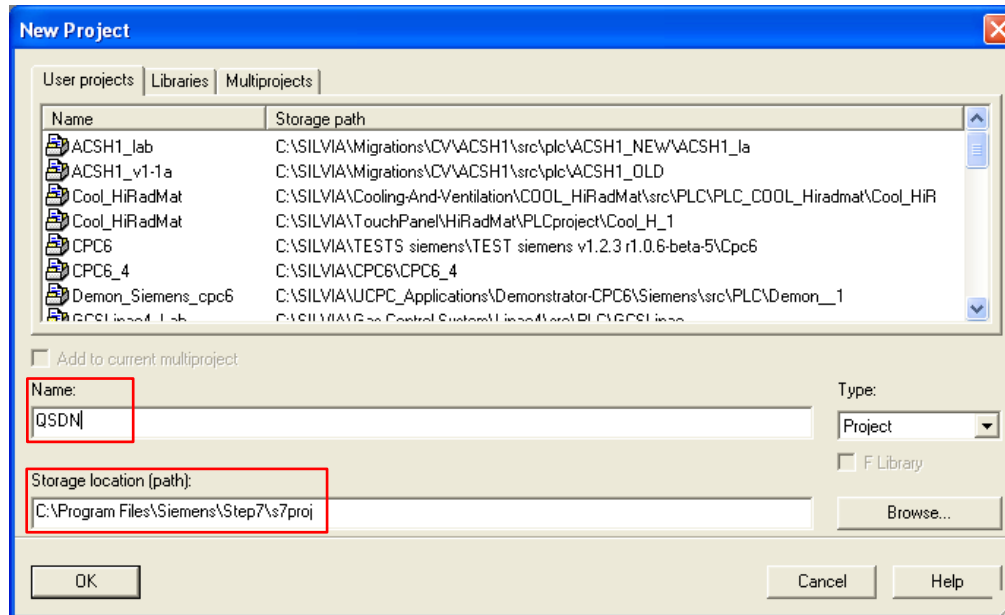
UAB User Report



```
[20/07/2011 10:35:48][INFO][PROGRAM] The instance files have been generated in: C:\SILVIA\TESTS\siemens\TEST\siemens_15072011\Output\S7CodeGenerator\  
[20/07/2011 10:35:48][INFO][PROGRAM] The log file has been generated in: C:\SILVIA\TESTS\siemens\TEST\siemens_15072011\Log\  
[20/07/2011 10:35:48][INFO][PROGRAM] DigitalInput objects: 6  
[20/07/2011 10:35:48][INFO][PROGRAM] AnalogInput objects: 6  
[20/07/2011 10:35:48][INFO][PROGRAM] DigitalParameter objects: 0  
[20/07/2011 10:35:48][INFO][PROGRAM] WordParameter objects: 0  
[20/07/2011 10:35:48][INFO][PROGRAM] AnalogParameter objects: 12  
[20/07/2011 10:35:48][INFO][PROGRAM] AnalogInputReal objects: 0  
[20/07/2011 10:35:48][INFO][PROGRAM] ProcessControlObject objects: 1  
[20/07/2011 10:35:48][INFO][PROGRAM] Controller objects: 2  
[20/07/2011 10:35:48][INFO][PROGRAM] DigitalAlarm objects: 3  
[20/07/2011 10:35:48][INFO][PROGRAM] AnalogAlarm objects: 1  
[20/07/2011 10:35:48][INFO][PROGRAM] Analog objects: 2  
[20/07/2011 10:35:48][INFO][PROGRAM] AnalogDigital objects: 1  
[20/07/2011 10:35:48][INFO][PROGRAM] Local objects: 0  
[20/07/2011 10:35:48][INFO][PROGRAM] OnOff objects: 3  
[20/07/2011 10:35:48][INFO][PROGRAM] WordStatus objects: 0  
[20/07/2011 10:35:48][INFO][PROGRAM] AnalogStatus objects: 0  
[20/07/2011 10:35:48][INFO][PROGRAM] AnalogOutputReal objects: 0  
[20/07/2011 10:35:48][INFO][PROGRAM] DigitalOutput objects: 4  
[20/07/2011 10:35:48][INFO][PROGRAM] AnalogOutput objects: 2  
[20/07/2011 10:35:48][INFO][PROGRAM] Generation is finished  
[20/07/2011 10:35:48][INFO][PROGRAM] Total errors found: 0  
[20/07/2011 10:35:48][INFO][PROGRAM] Total warnings found: 7  
[20/07/2011 10:35:48][INFO][PROGRAM] The exit status of the S7CodeGenerator plug-in is SUCCESS.  
[20/07/2011 10:35:48][INFO][DATA] Stopping the plugin: S7CodeGenerator.  
[20/07/2011 10:35:48][INFO][DATA] Unplugging the plugin: S7CodeGenerator.  
[20/07/2011 10:35:48][INFO][PROGRAM] CoreManager.stop()  
[20/07/2011 10:35:48][INFO][PROGRAM] PluginManager.stop()  
[20/07/2011 10:35:48][INFO][PROGRAM] CoreManager.unplug()  
[20/07/2011 10:35:48][INFO][PROGRAM] CoreManager.shutdown()
```

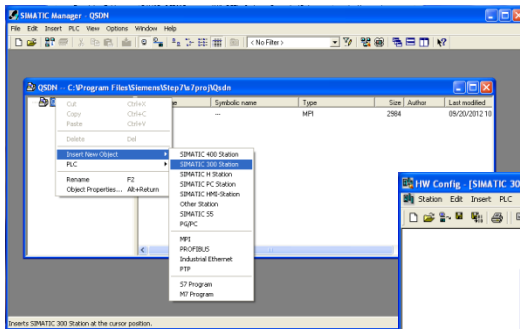
```
[20/07/2011 11:20:34][INFO][PROGRAM] The DB file generation is completed, 147 lines written to: C:\SILVIA\TESTS\siemens\TEST\siemens_15072011\Output\PersistenceGenerator\psst_db_file.txt  
[20/07/2011 11:20:34][INFO][PROGRAM] DigitalInput objects: 6  
[20/07/2011 11:20:34][INFO][PROGRAM] AnalogInput objects: 6  
[20/07/2011 11:20:34][INFO][PROGRAM] DigitalParameter objects: 0  
[20/07/2011 11:20:34][INFO][PROGRAM] WordParameter objects: 0  
[20/07/2011 11:20:34][INFO][PROGRAM] AnalogParameter objects: 12  
[20/07/2011 11:20:34][INFO][PROGRAM] AnalogInputReal objects: 0  
[20/07/2011 11:20:34][INFO][PROGRAM] ProcessControlObject objects: 1  
[20/07/2011 11:20:34][INFO][PROGRAM] Controller objects: 2  
[20/07/2011 11:20:34][INFO][PROGRAM] DigitalAlarm objects: 3  
[20/07/2011 11:20:34][INFO][PROGRAM] AnalogAlarm objects: 1  
[20/07/2011 11:20:34][INFO][PROGRAM] Analog objects: 2  
[20/07/2011 11:20:34][INFO][PROGRAM] AnalogDigital objects: 1  
[20/07/2011 11:20:34][INFO][PROGRAM] Local objects: 0  
[20/07/2011 11:20:34][INFO][PROGRAM] OnOff objects: 3  
[20/07/2011 11:20:34][INFO][PROGRAM] WordStatus objects: 0  
[20/07/2011 11:20:34][INFO][PROGRAM] AnalogStatus objects: 0  
[20/07/2011 11:20:34][INFO][PROGRAM] AnalogOutputReal objects: 0  
[20/07/2011 11:20:34][INFO][PROGRAM] DigitalOutput objects: 4  
[20/07/2011 11:20:34][INFO][PROGRAM] AnalogOutput objects: 2  
[20/07/2011 11:20:34][INFO][PROGRAM] Generation is finished  
[20/07/2011 11:20:34][INFO][PROGRAM] Total errors found: 3  
[20/07/2011 11:20:34][INFO][PROGRAM] Total warnings found: 0  
[20/07/2011 11:20:34][SEVERE][PROGRAM] The exit status of the PersistenceCodeGenerator plug-in is FAILURE.  
[20/07/2011 11:20:34][INFO][DATA] Stopping the plugin: PersistenceCodeGenerator.  
[20/07/2011 11:20:34][INFO][DATA] Unplugging the plugin: PersistenceCodeGenerator.  
[20/07/2011 11:20:34][INFO][PROGRAM] CoreManager.stop()  
[20/07/2011 11:20:34][INFO][PROGRAM] PluginManager.stop()  
[20/07/2011 11:20:34][INFO][PROGRAM] CoreManager.unplug()  
[20/07/2011 11:20:34][INFO][PROGRAM] CoreManager.shutdown()
```


Create a new S7 project

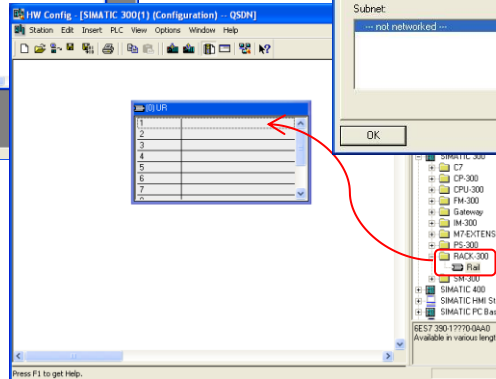


1. PLC configuration
 - a) Hardware

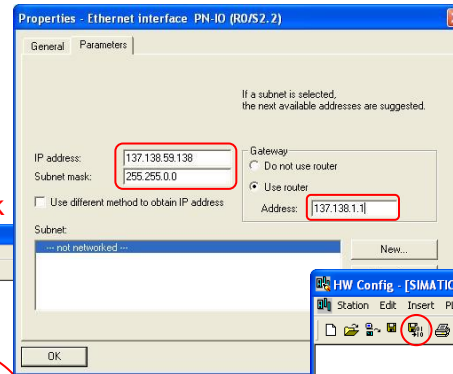
1. Insert 300 station



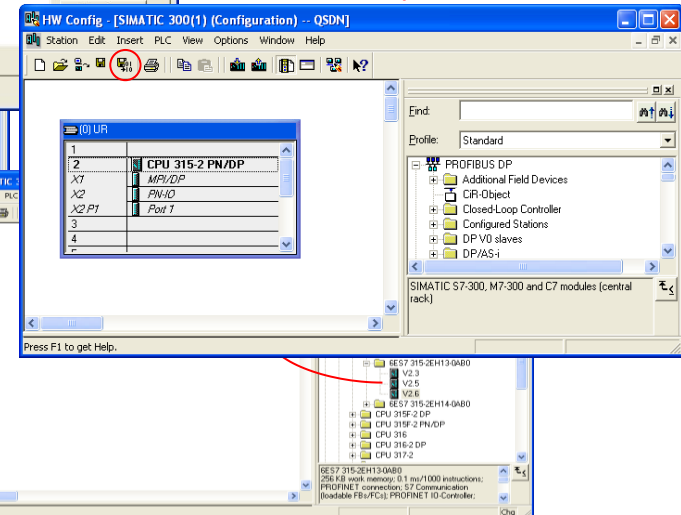
2. Insert rack



4. Configure CPU properties

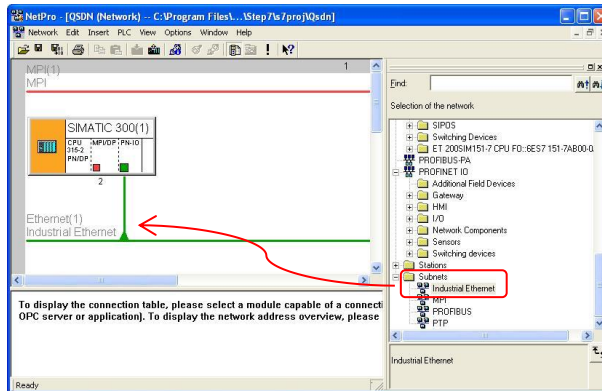


5. Save and compile

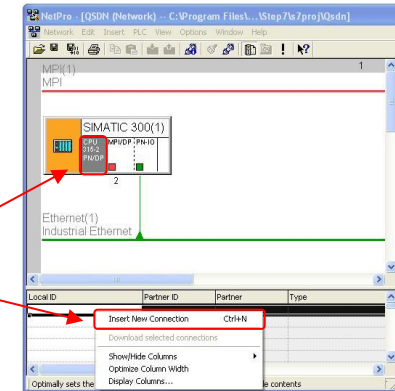


b) Connection

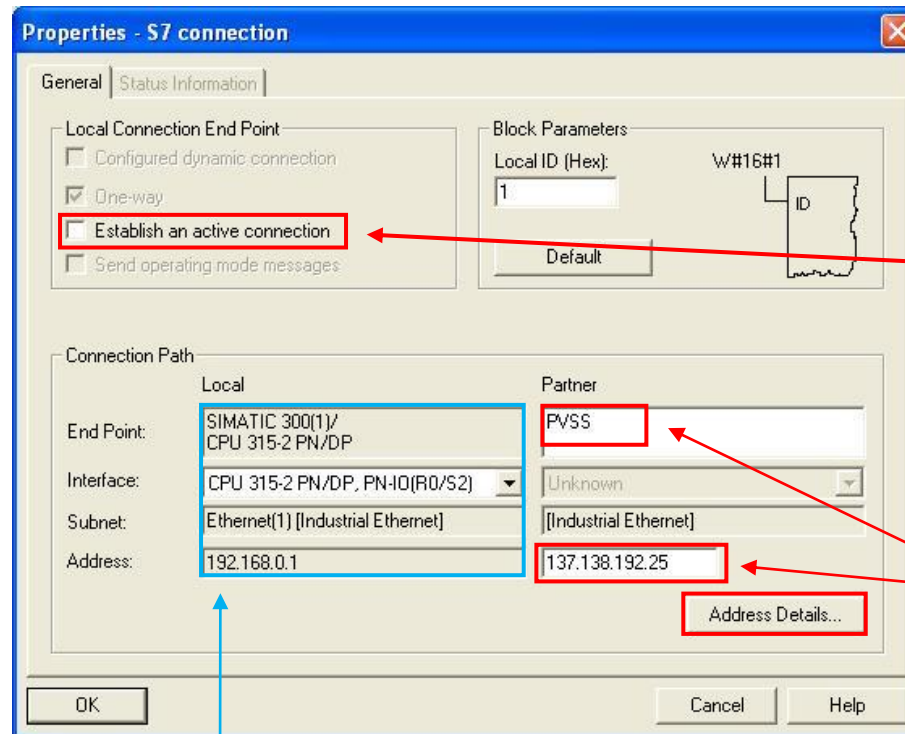
1. Insert Industrial Ethernet



2. Insert a new connection



3. Set the S7 connection properties for the supervision



The connection must **NOT** be active. Uncheck this box

The partner must be the PVSS machine with its IP number

PLC information

4. Fill in the different fields according to the hardware which has been specified for your application

Address Details

	Local	Partner
End Point:	SIMATIC 300(1) CPU 315-2 PN/DP	PVSS
Rack/Slot:	0 2	0 0
Connection Resource (hex):	10	10
TSAP:	10.02	10.00
S7 Subnet ID:	0000 - 0007	-

Buttons: OK, Cancel, Help

Rack/Slot and Connection Resource of PLC

Rack/Slot and Connection Resource of WinCC OA

CPC Wizard: QSDN - QSDN v1.0
SIEMENS PLC Specifications
Resources: 1.6.0

General Data

PLC Name: CFP_LABO_BEN
IP Address: 137.138.192.1
PLC Type: S7-300 PN/DP

Recipes

Enable recipes:
Max. number of recipe values: 1000
Activation Timeout (s): 100

WinCC OA Configuration

Partner Rack: 0
Partner Slot: 0
Partner Con. Resource: 10
Timeout: 5000

PLC Connection

Local Id: 1
Local Rack: 0
Local Slot: 2
Local Con. Resource: 10

User Resources Configuration

Dynamic User Resources File: Browse Clear
Static User Resources File: Browse Clear

General Configuration

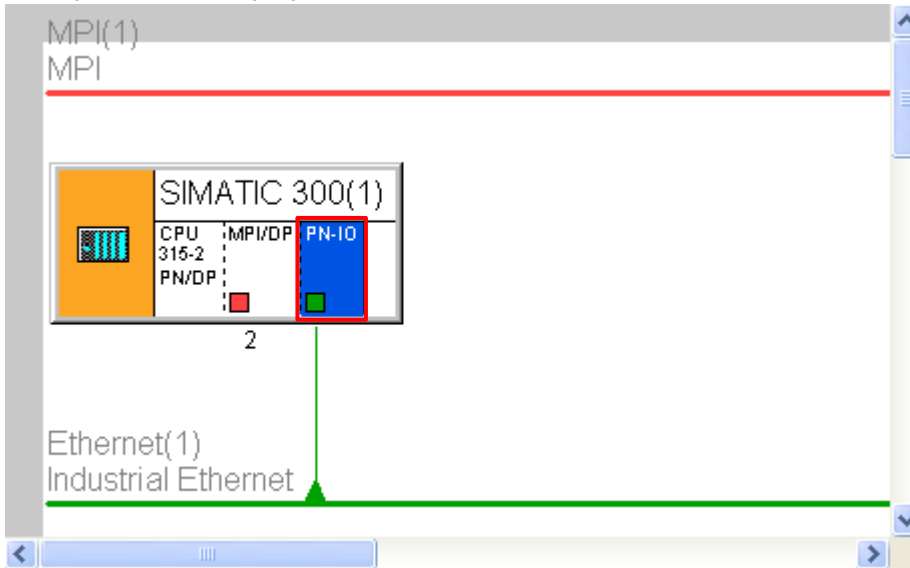
Diagnostic Channel:
Event Buffer Size: 1000

Address Configuration

1st FB: 100
1st FC: 100
1st DB: 100
1st UDT: 100

Buttons: Back, Next, Exit

3. Open the PN-IO properties



137.138.16.69

137.138.17.69

4. Enable and add the NTP Server Addresses

Properties - PN-IO (R0/S2.2)

General | Addresses | PROFINET | Synchronization | Time-of-Day Synchronization

NTP Mode
 Enable Time-of-Day Synchronization in NTP Mode

NTP Server Addresses:

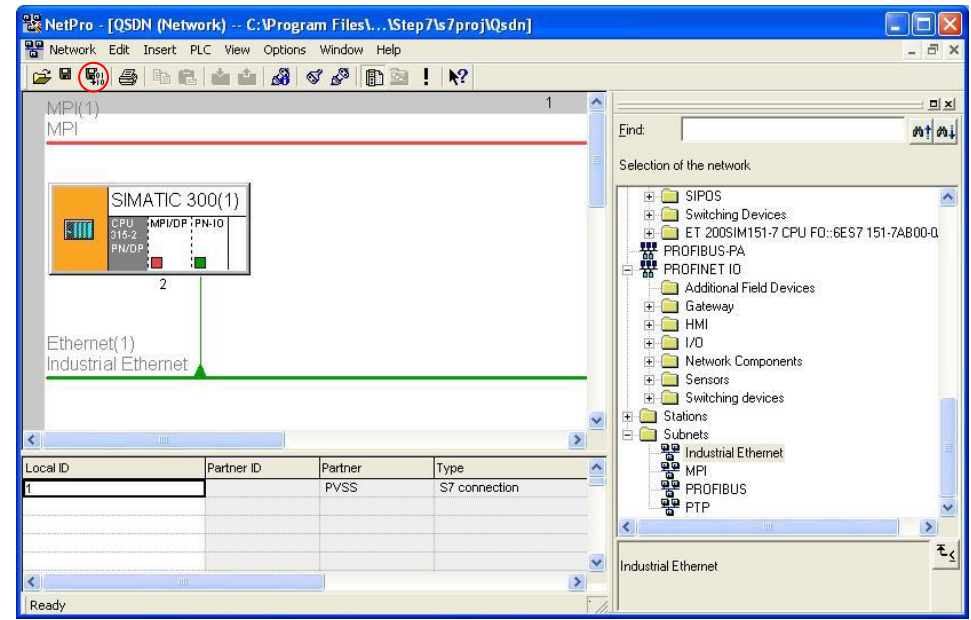
172.18.16.69
172.18.17.69

Add...
Edit...
Delete

Update Interval [Seconds]:
[Value Range 10-86400] 10

OK Cancel Help

5. Save and compile



The parameters chosen have to match with the ones used in the UAB CPC wizard

The image displays three screenshots of Siemens software configuration windows, illustrating the consistency of parameters across different stages of the configuration process:

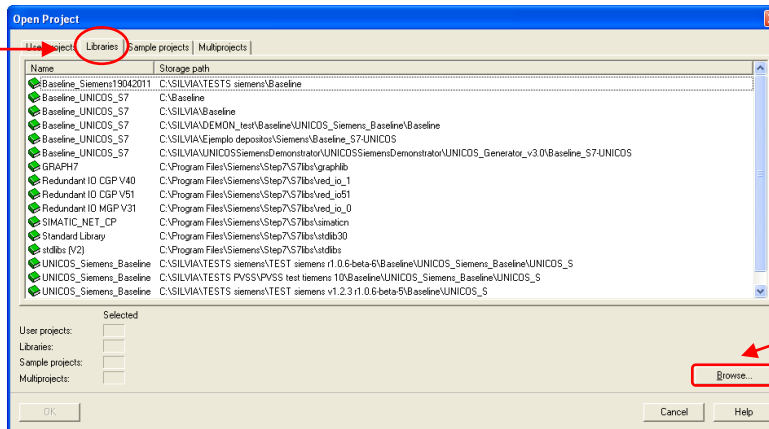
- UAB CPC Wizard v1.3.2-beta-01:** Shows the main configuration screen. Key parameters highlighted with red boxes include:
 - IP Address: 137.138.192.1
 - Partner Rack: 0
 - Partner Slot: 0
 - Partner Con. Resource: 10
 - Local Rack: 0
 - Local Slot: 2
 - Local Con. Resource: 10
- Properties - S7 connection:** Shows the connection details. The Local Address is highlighted with a red box and matches the IP address in the wizard: 137.168.0.1.
- Address Details:** Shows the rack and slot configuration. The Rack/Slot (0 2) and Connection Resource (10) are highlighted with red boxes, matching the Partner and Local connection resources in the wizard.

Red arrows indicate the flow of information and consistency between these windows, showing how the parameters defined in the wizard are reflected in the connection and address details.

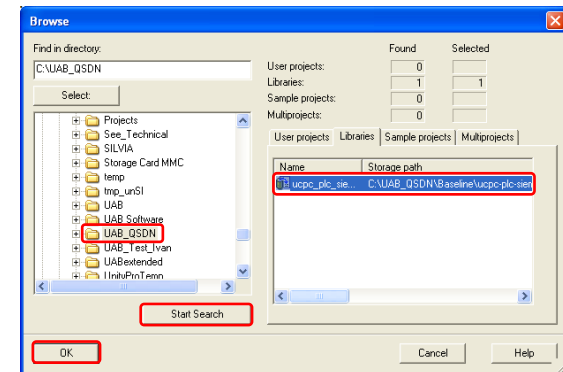
Importation into Siemens S7

1. Baseline importation

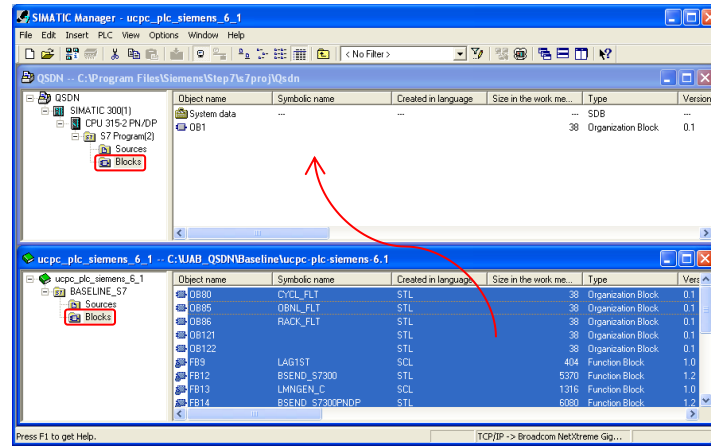
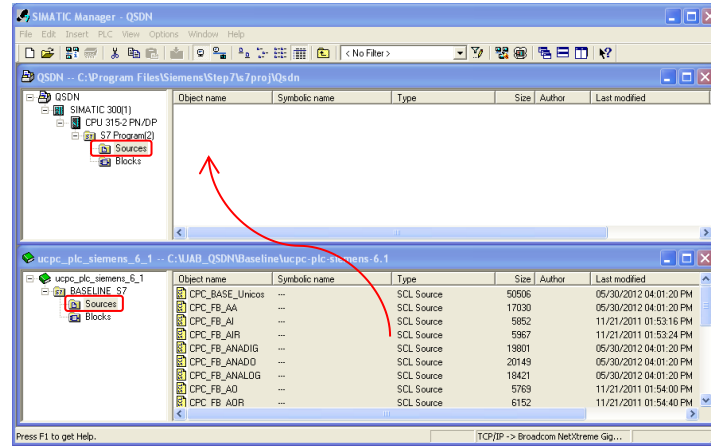
Select libraries



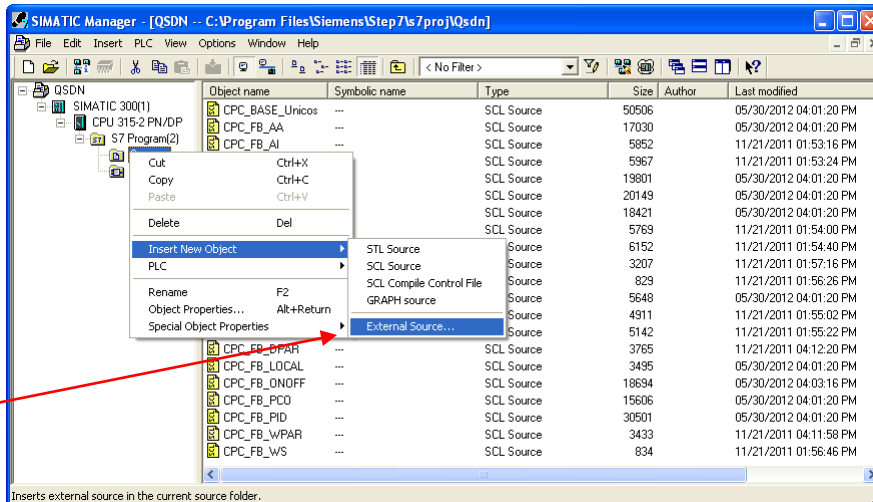
Browse UNICOS baseline in the project folder



Creation of a Siemens S7 UNICOS-CPC 6 application

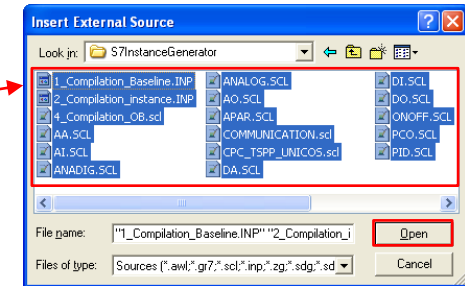


2. Instance importation



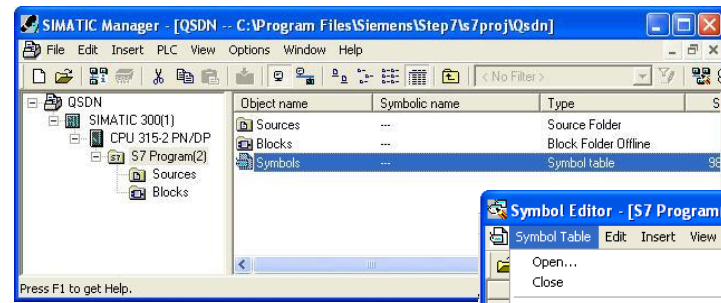
Insert the instance sources

Select all the instance files

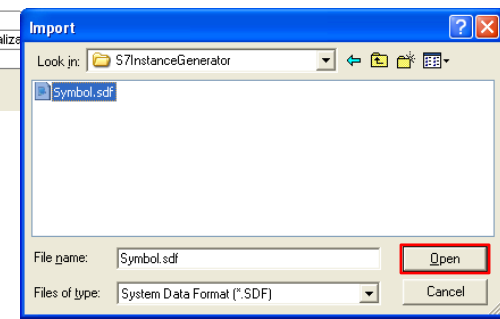
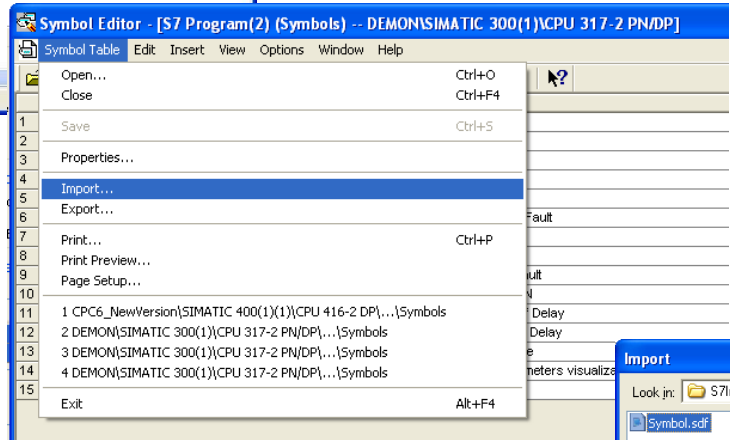


3. Symbol table importation

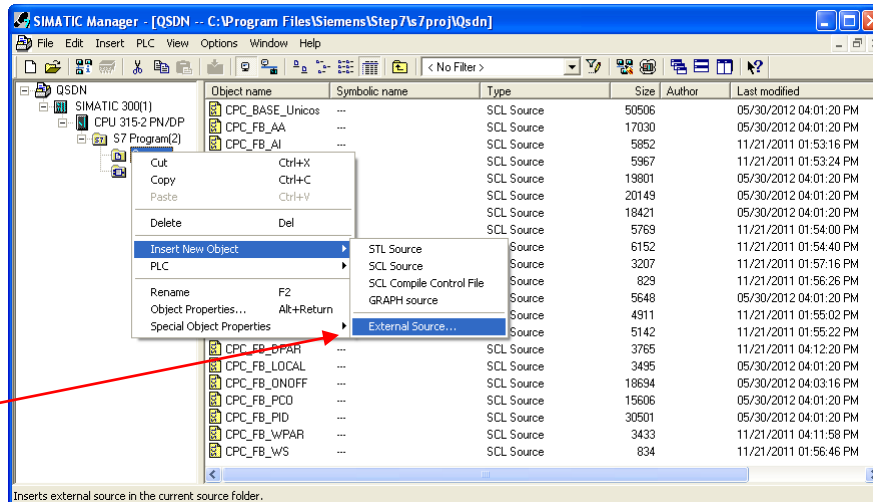
1. Open the symbol table



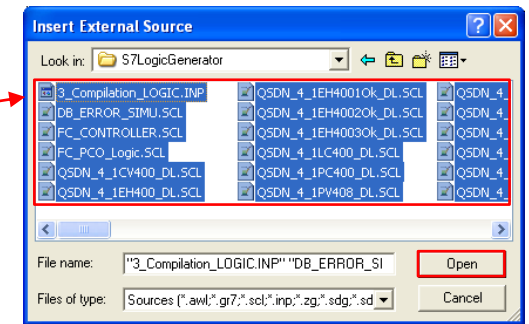
2. Import the symbol table generated



4. Logic importation



Insert the logic sources



Select all the logic files

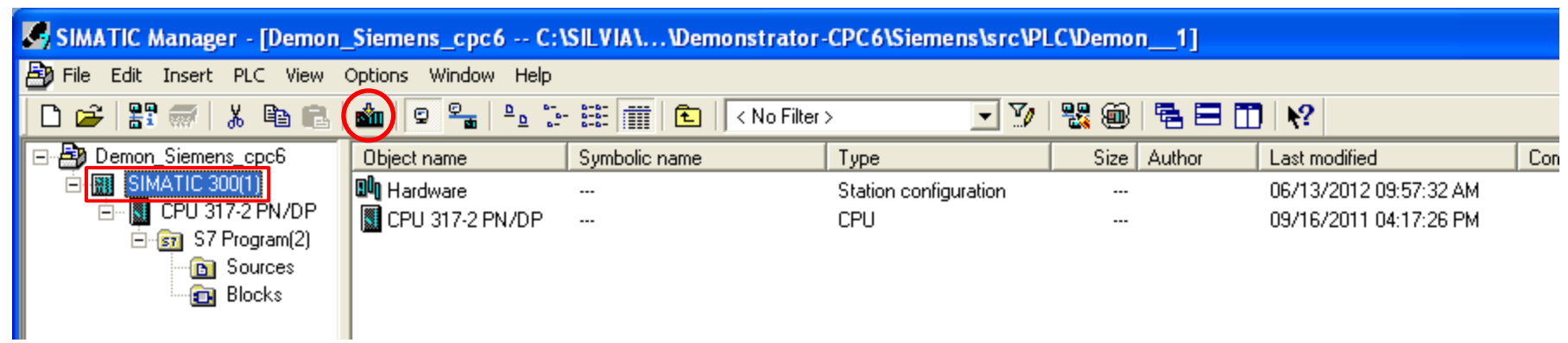
4. Compilation

The image displays four overlapping windows from the SIMATIC Manager software, each showing the compilation process for a different SCL file. The windows are titled as follows:

- [1_Compilation_Baseline -- QSDNSIMATIC 300(1)\CPU 315-2 PN/DP]**: Shows the compilation of the baseline file. The code includes comments like "(*Launch of the SCL files*****" and lists various data types and functions such as `CPC_BASE_Unicos;`, `CPC_TSPP_Unicos;`, and `CPC_FE_DI;`.
- [2_Compilation_instance -- QSDNSIMATIC 300(1)\CPU 315-2 PN/DP]**: Shows the compilation of the instance file. The code includes comments like "(*Launch of the SCL files*****" and lists various data types and functions such as `DI;`, `AI;`, `AIR;`, `DO;`, `AOR;`, `DPAR;`, `WPAR;`, `APAR;`, `WS;`, `AS;`, `LOCAL;`, `ONOFF;`, `ANALOG;`, `ANADIG;`, `PID;`, `PCO;`, `DA;`, `AA;`, and `COMMUNICATION;`.
- [3_Compilation_LOGIC -- QSDNSIMATIC 300(1)\CPU 315-2 PN/DP]**: Shows the compilation of the logic file. The code includes comments like "(*Launch of the SCL files*****" and lists various data types and functions such as `DB_ERROR_SIMU;`, `QSDN_4_DNCT_IL;`, `QSDN_4_DNCT_CL;`, `QSDN_4_DNCT_BL;`, `QSDN_4_DNCT_INST;`, `QSDN_4_DNCT_GL;`, `QSDN_4_DNCT_TL;`, `QSDN_4_DNCT_SL;`, `QSDN_4_DNCT_CDOL;`, `QSDN_4_ESSCRst_DL;`, `QSDN_4_CV400_DL;`, `QSDN_4_DN1CT_IL;`, `QSDN_4_DN1CT_CL;`, `QSDN_4_DN1CT_BL;`, `QSDN_4_DN1CT_INST;`, `QSDN_4_DN1CT_GL;`, `QSDN_4_DN1CT_TL;`, `QSDN_4_DN1CT_SL;`, `QSDN_4_DN1CT_CDOL;`, `QSDN_4_DN1CT_DL;`, `QSDN_4_1EH40010K_DL;`, `QSDN_4_1EH40020K_DL;`, `QSDN_4_1EH40030K_DL;`, and `QSDN_4_1PV408_DL;`.
- [4_Compilation_OB -- QSDNSIMATIC 300(1)\CPU 315-2 PN/DP]**: Shows the compilation of the OB file. The code includes comments like "(*Launch of the SCL files*****" and lists various data types and functions such as `OB100_EV_CLASS`, `OB100_STRUP`, `OB100_PRIORITY`, `OB100_OB_NUMBER`, `OB100_RESERVED_1`, `OB100_RESERVED_2`, `OB100_STOP`, `OB100_STRT_INFO`, and `OB100_DATE_TIME`.

A red arrow points to the 'C' icon in the toolbar of the first window, labeled "Compilation".

Downloading the application to the PLC

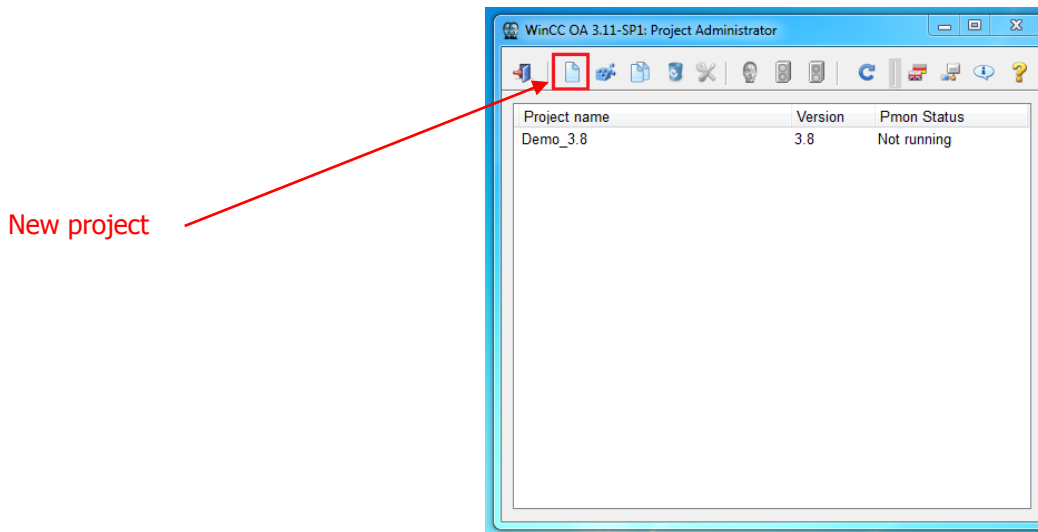




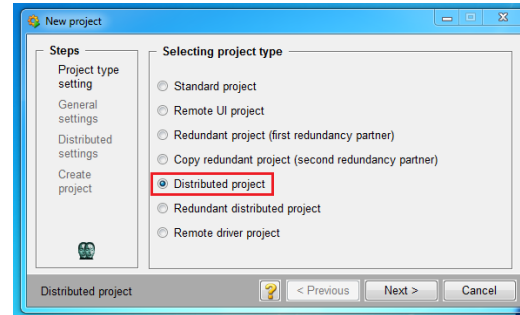
Creation of a WinCC OA UNICOS-CPC 6 application

Creating a WinCC OA project

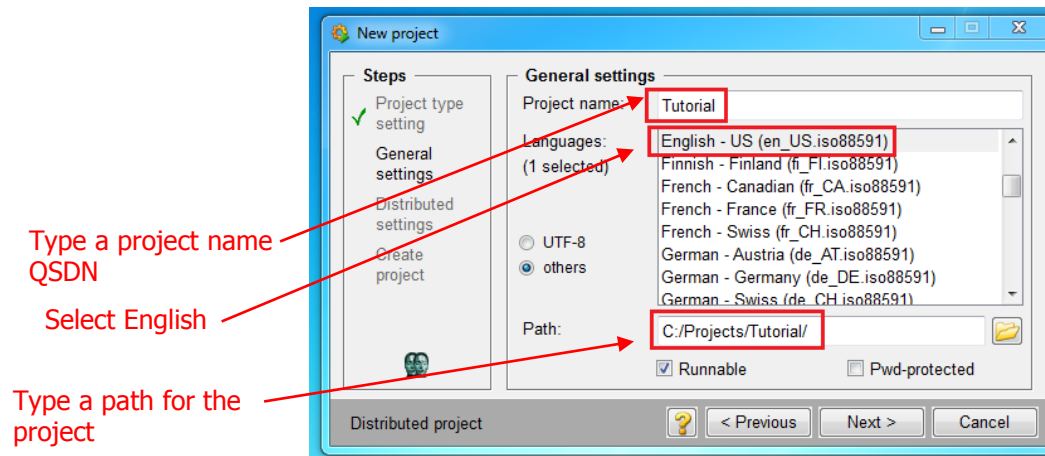
1. Start the WinCC OA Project Administration with administrator privileges
2. Click on "New Project"



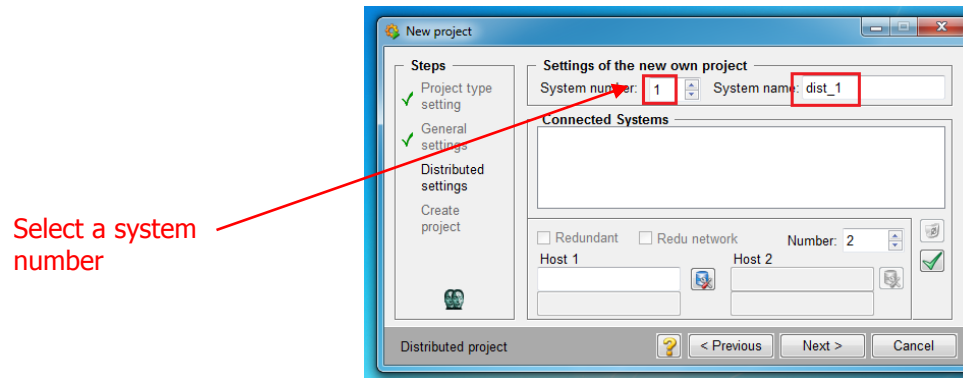
3. Set the project as "Distributed project"



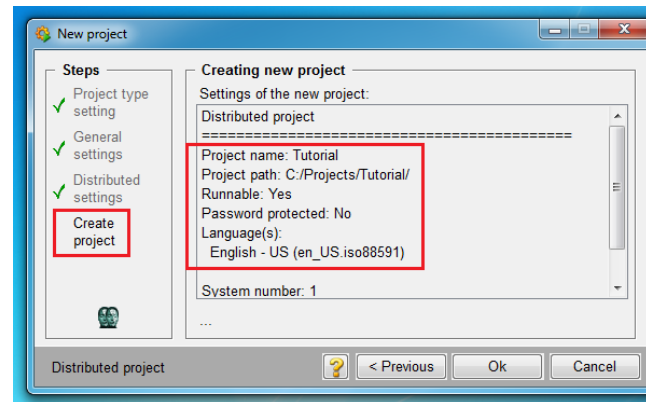
4. Set the general settings



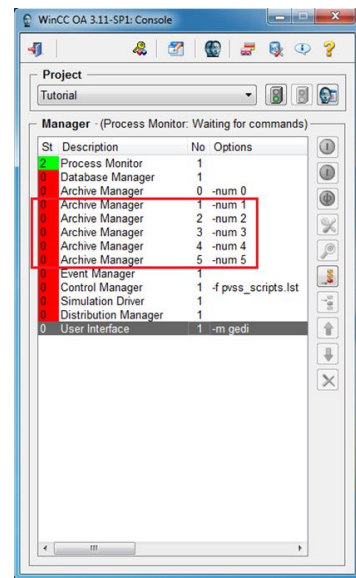
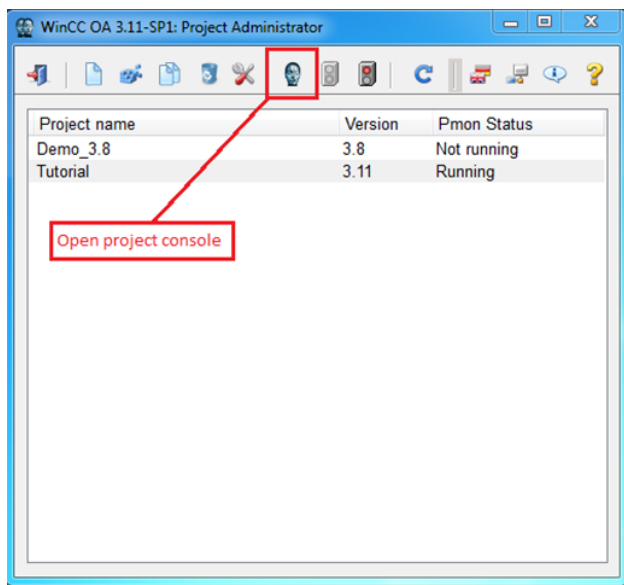
5. Set the "Distributed settings"



6. Check all settings and click "OK"

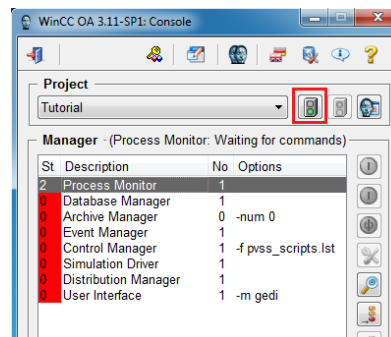


- 7. Launch the project console and remove Archive managers from 1 to 5

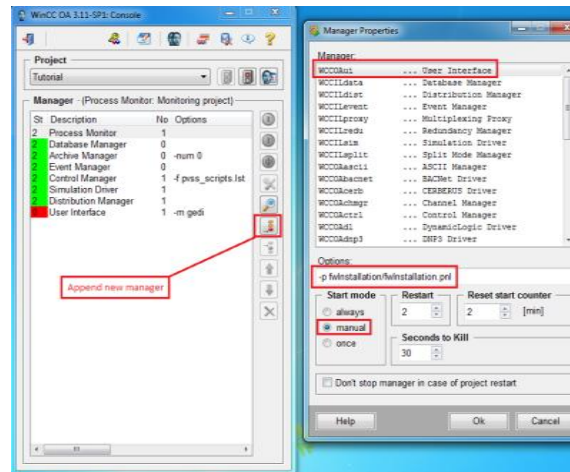


Framework component installation

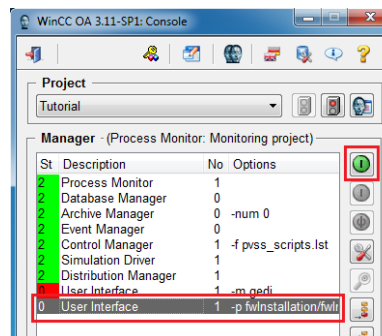
1. Unzip the Framework Component Installation package on top of the project directory (fw-installation-tool)
2. Start WinCCOA project



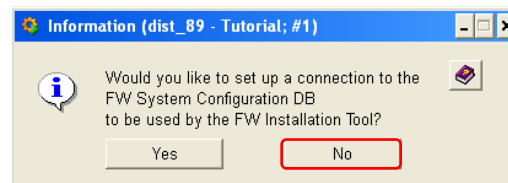
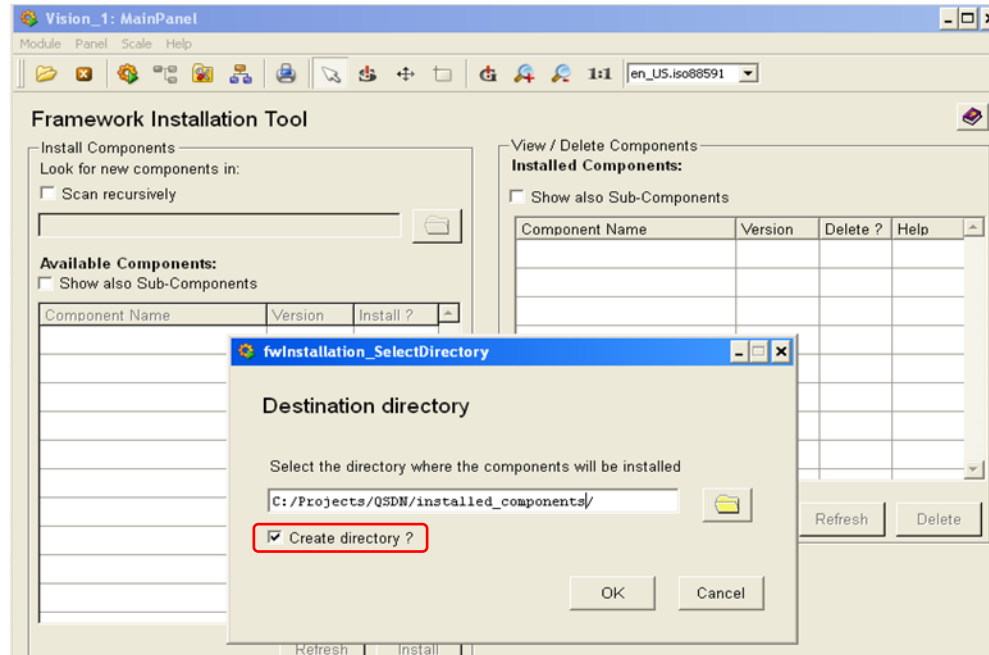
3. Create the user interface for the installation with the following configuration: `-p fwInstallation/fwInstallation.pnl`



3. Run the User Interface created for the installation



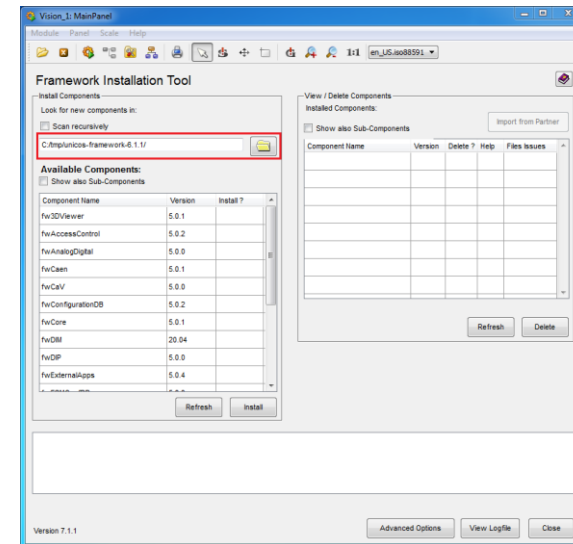
5. Choose a destination directory



Installation of UNICOS packages

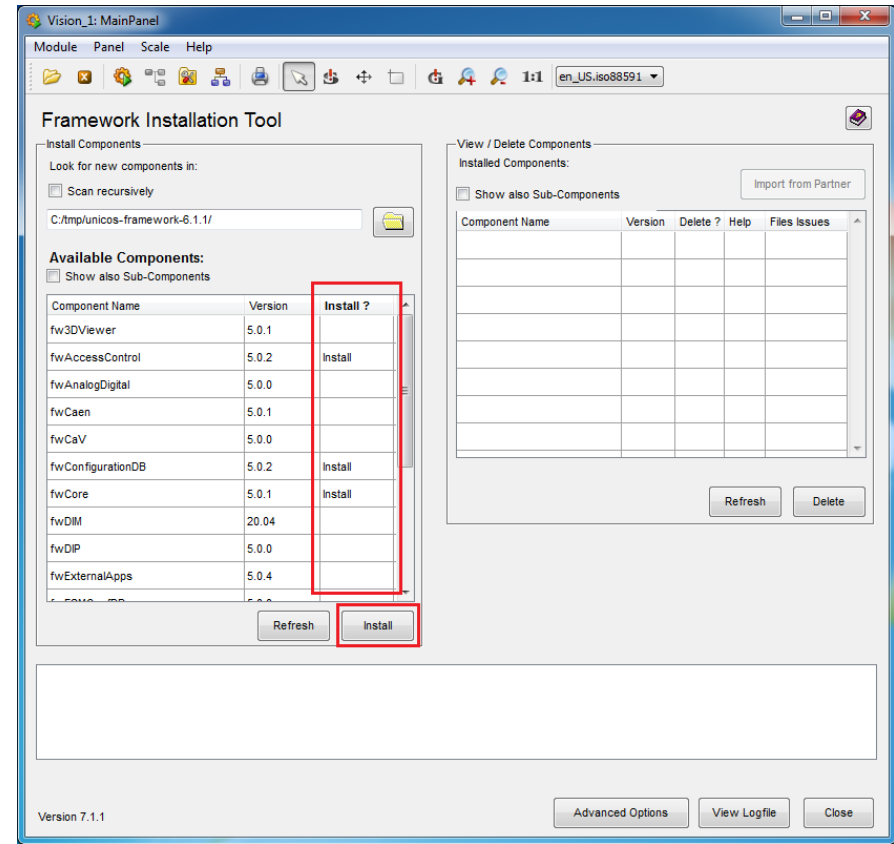
Installation of CORE packages

1. Unzip the UNICOS Core package (unicos-framework-winccoa) in a temporary location
2. Select the directory where the package has been unzipped in the Framework Installation Tool

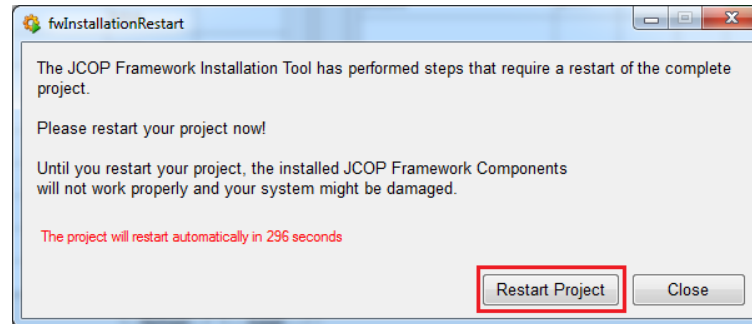


3. Install the following components

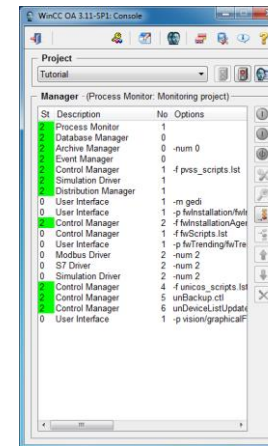
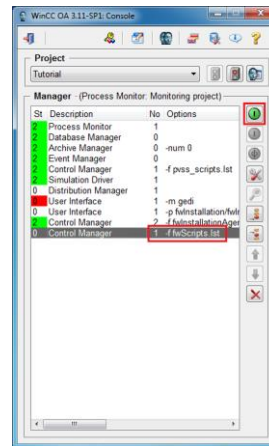
- fwAccessControl
- fwCore
- fwTrending
- unCore
- fwConfigurationDB



4. Restart the project



5. Launch fwScripts.lst



Installation of CPC package

1. Unzip the UNICOS CPC package in a temporary location (ucpc-wincc-oa)
2. Select the directory where the package has been unzipped in the Framework Installation Tool
3. Start the User Interface (fwInstallation) Manager in the console
4. Install the components
 - unCPC6
 - unRecipe
5. Restart the project
6. Launch fwScripts.lst

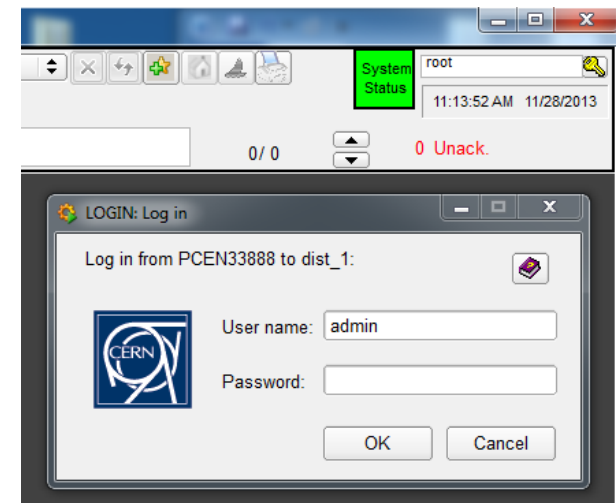
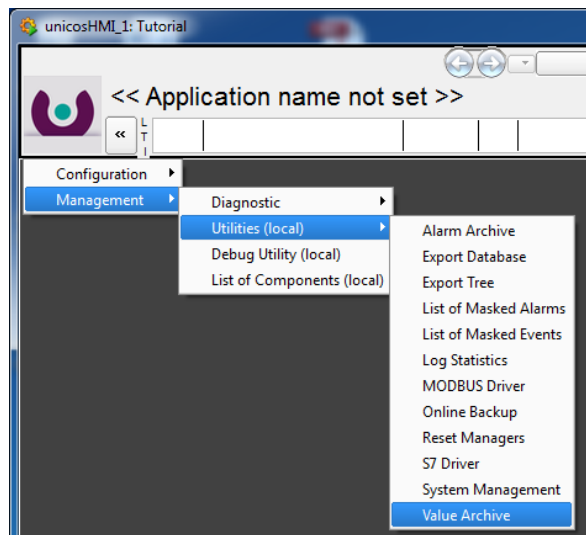
Setting up archives

1. Launch HMI by launching the User Interface

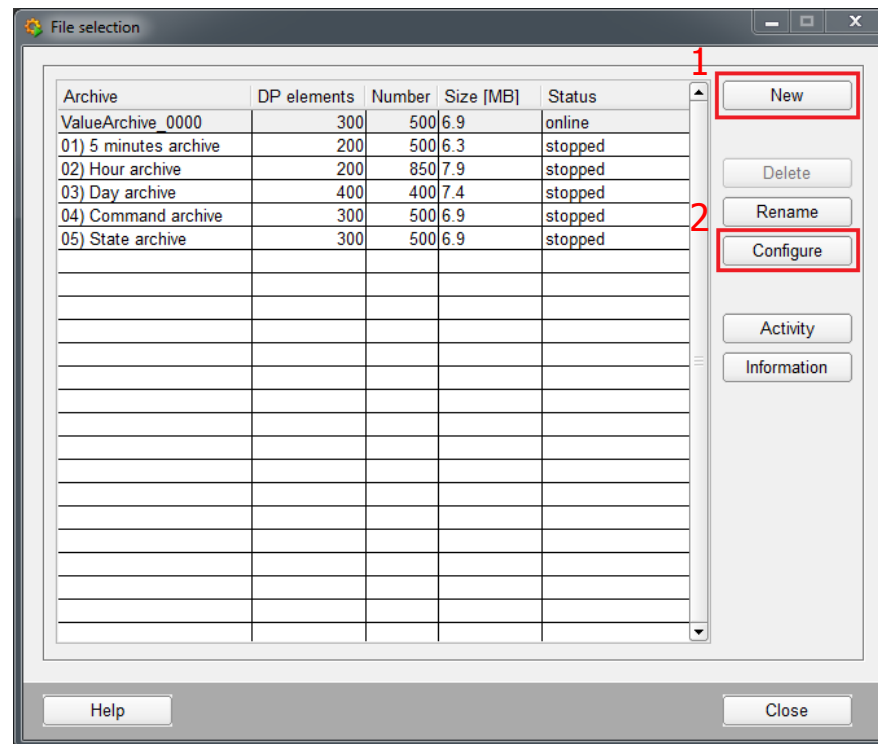
“-p vision/graphicalFrame/unicosHMI.pnl”

2. Log in as “admin” (no password)

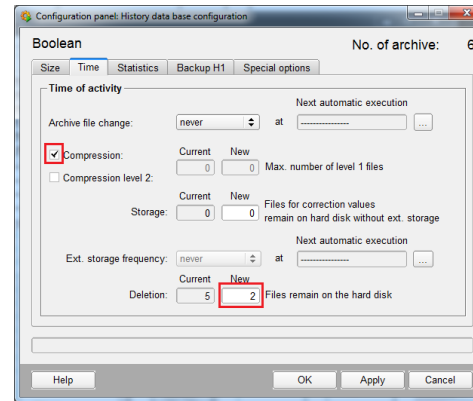
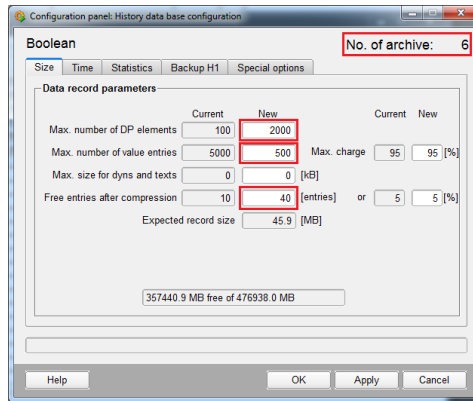
3. Open “Value Archive”



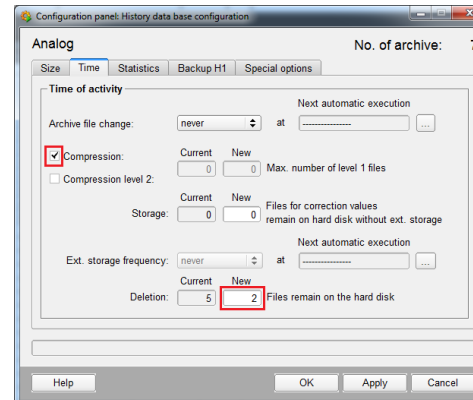
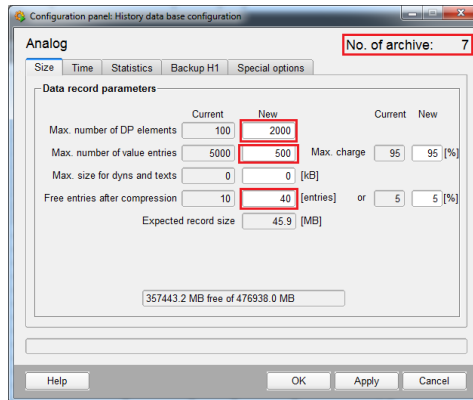
4. Create a "boolean", "analog" and "event" archive



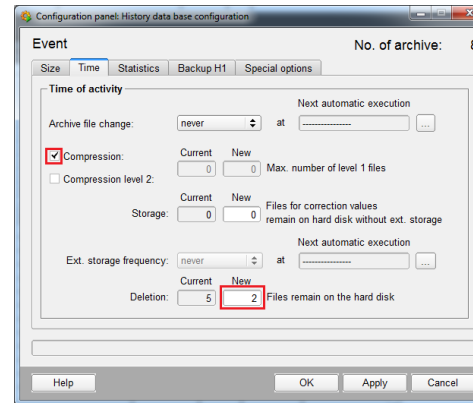
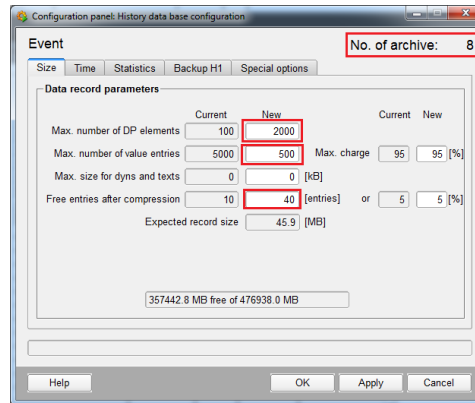
Parameterize the Boolean archive



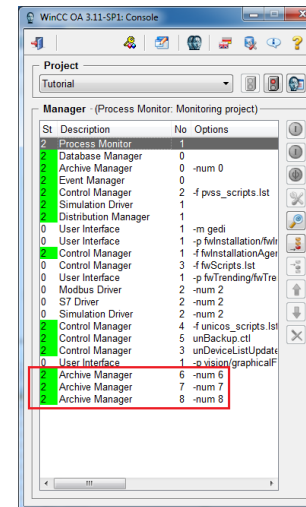
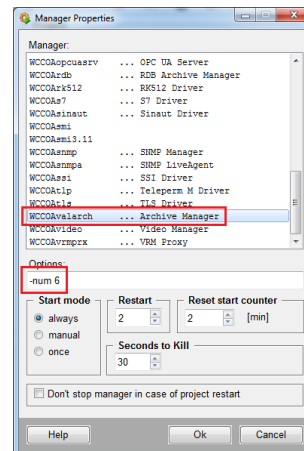
Parameterize the Analog archive



Parameterize the Event archive

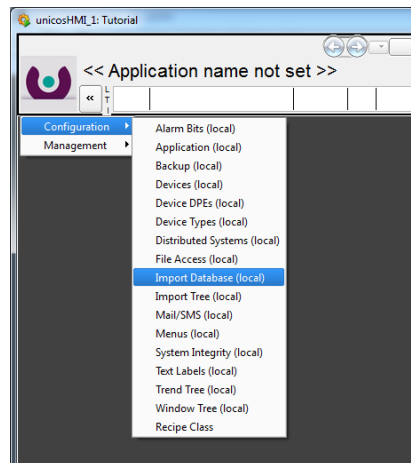


5. Create corresponding archive managers in the PVSS console



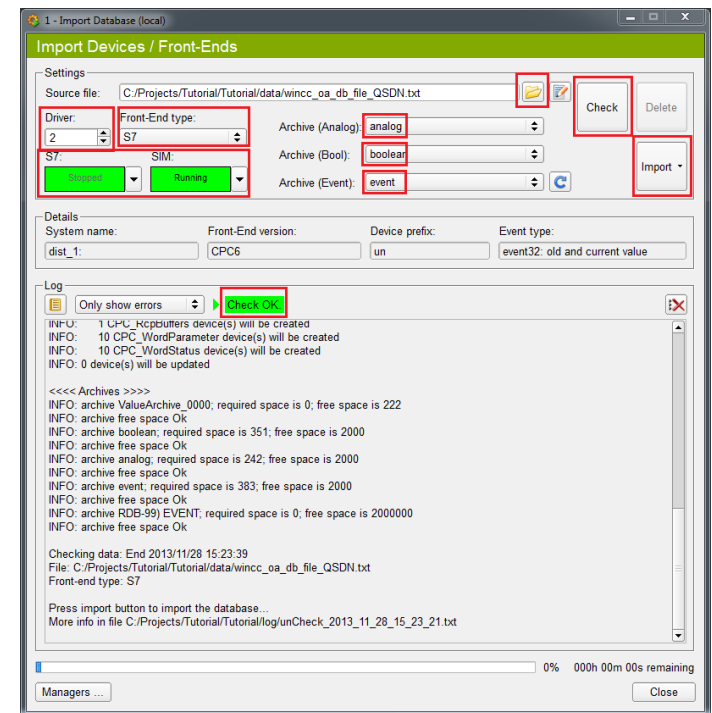
Importation into PVSS

1. Start the unicosHMI.pnl
(-p vision/graphicalFrame/unicosHMI.pnl)
2. Log in as "admin" and start the "Import Database" panel

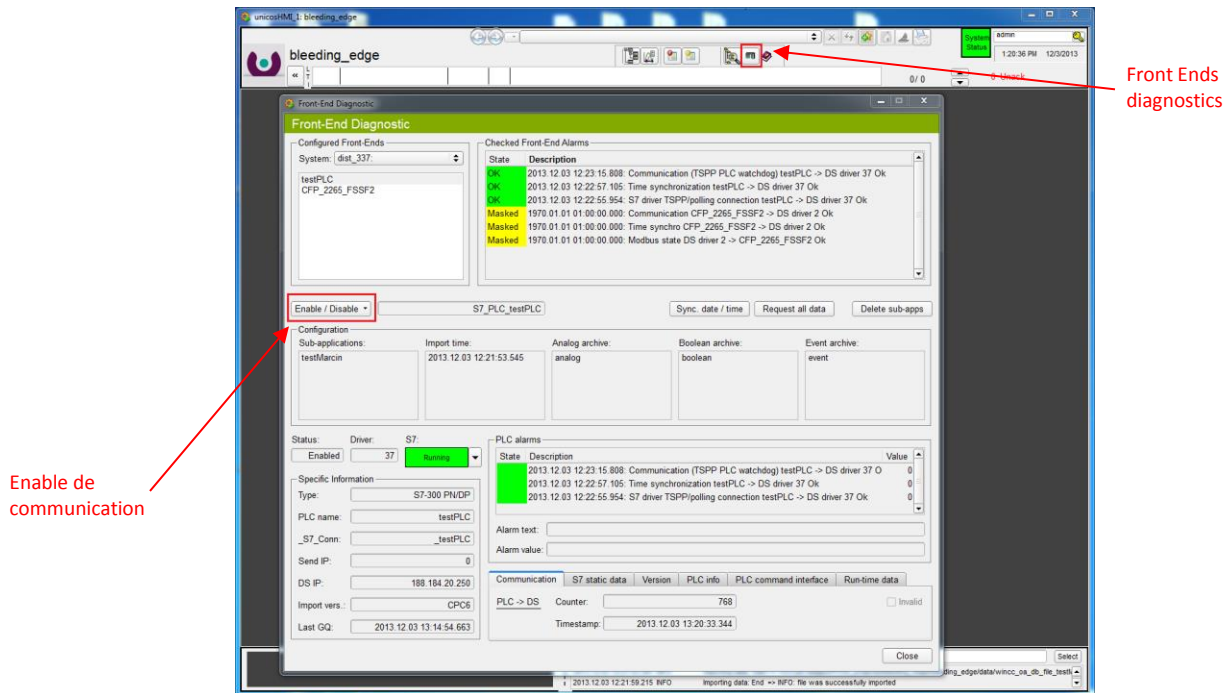


3. Stop Distribution Driver (change it from always to manual)
4. Copy the importation file to the PVSS project
(C:\Projects\QSDN\QSDN\data)

5. Select the proper Driver Number
6. Set the Front End type to “_S7Plc”
7. Stop the S7 Driver and start the Simulation Driver
8. Select the archive class for Bool, Analog and Event
9. Select the importation file generated
(copy it to
C:\Projects\QSDN\QSDN\data)
10. Do a check and import



9. Stop the Simulation Driver and start the S7 Driver
10. Open the Front Ends Diagnostic and enable the communication



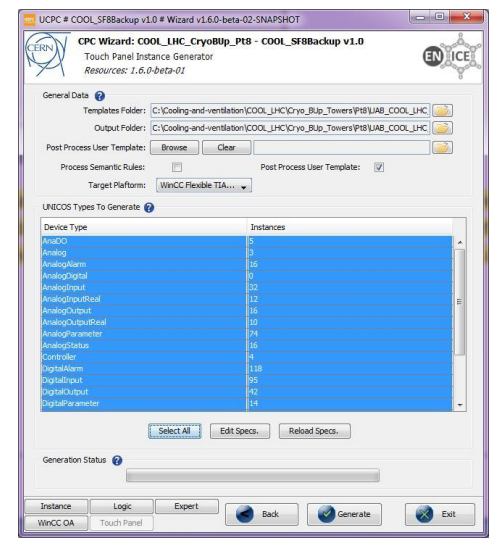
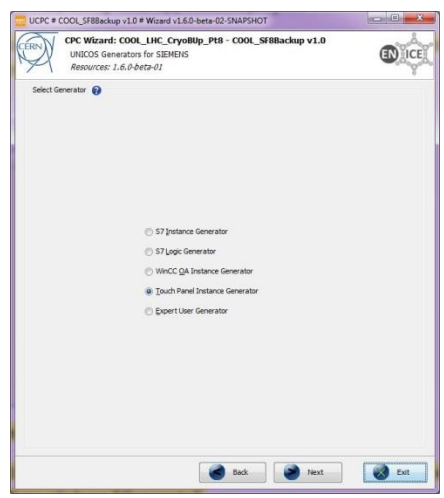


Creation and design of a local panel for a UNICOS CPC6 application

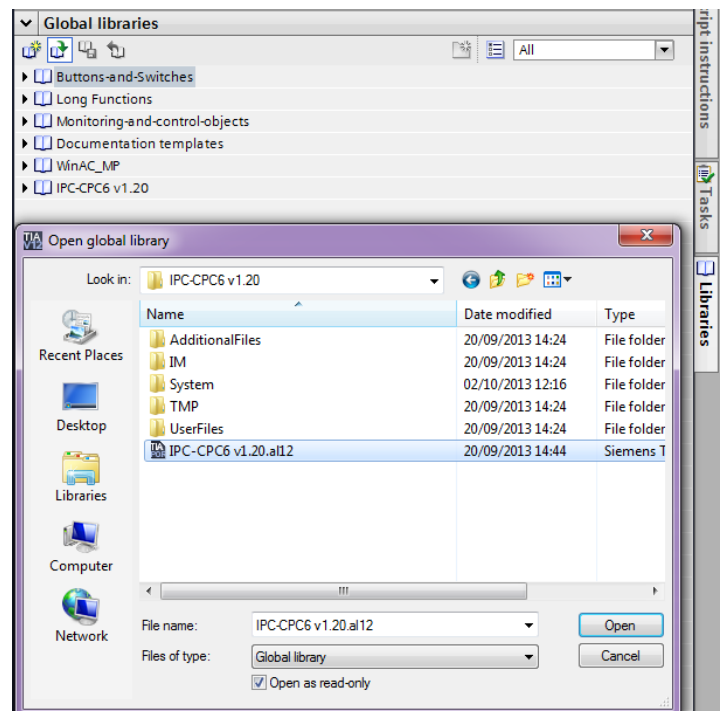
Creation and design of a local panel for a UNICOS CPC6 application

UAB generation for Touch panels

- Alarms
- Scripts
- Tags
- Text lists



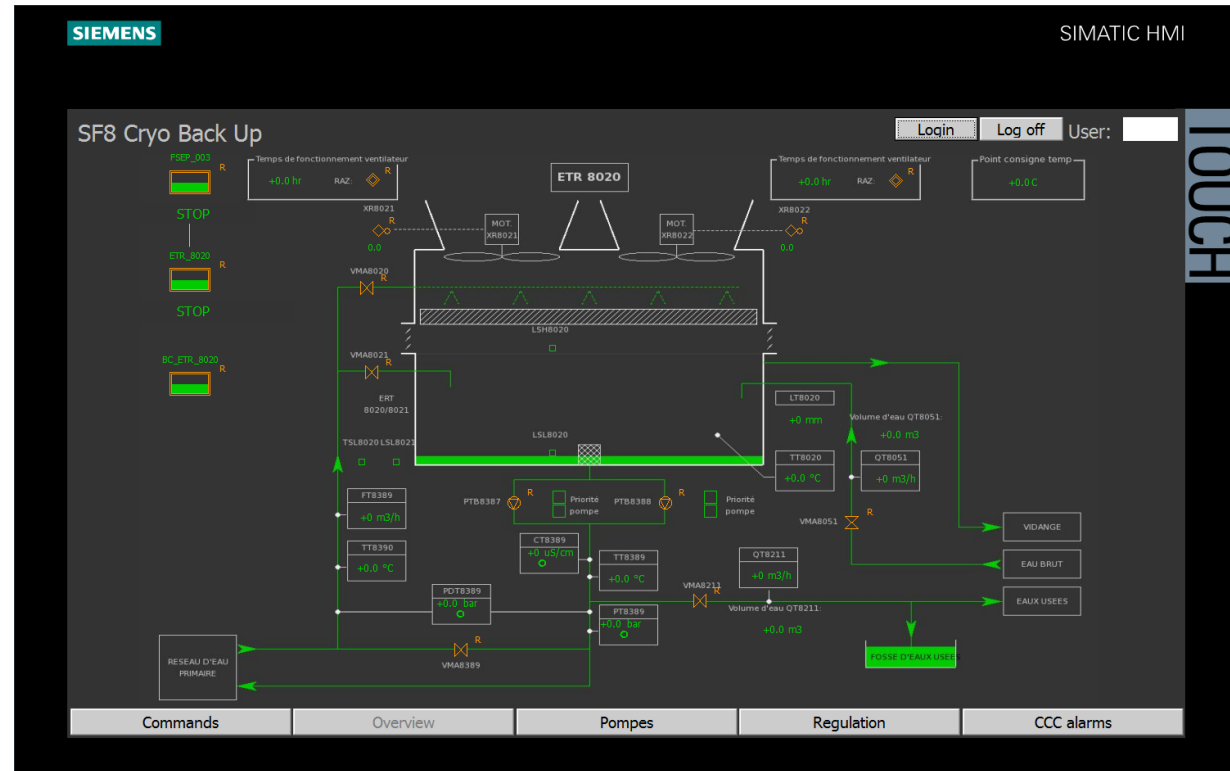
UNICOS library for touch panels



Creation and design of a local panel for a UNICOS CPC6 application

Touch Panel design

"...\UAB_Project\Baseline\ucpc-wincc-tia-vx.x"



Connection

Type of communication TP-PLC: Ethernet

