Process control system developed with SoMachine Platform and UNICOS CPC framework.
TUTORIAL GOALS

This video tutorial presents you all steps necessary to achieve rapidly a CPC project based on the SoMachine development platform.

This video is not a training on SoMachine, CoDeSys or UCPC.

With the video, you will get real examples and observe all problems that you have usually to face when you use a new product.

I want to thanks all people who have contribute to the project.

Karolina MOKRZYSZ  concept validation CoDeSys V2.3
David WILLEMAN  Support, Communication drive
Theodora PETROU  Implementation SoMachine (CoDeSys V3)
Maria KOUTLI  Implementation SoMachine/TwinCAT
Ivan PRIETO BARRIERO  Support, UAB, UCPC

https://wikis.web.cern.ch/wikis/display/EN/CoDeSys+-+SoMachine
The tutorial is composed.

• Introduction
• Definitions of tools and major terms.
• Description of Schneider SoMachine PLCs
• Installation/preparation of the SoMachine platform
• IP address registration.
• Preparation of a CPC SoMachine Baseline.
• Connection to the hardware PLC.
• Project generation with UAB.
• Project build and downloading in the PLC.
**PLC DEFINITION.**

**PLC** (Programmable Logic Controller), was until the 1990, executing a set of sequences introduced via a terminal. With the computing emerging period, it has migrated to a computer. For security reason, its software and implementation were build mainly with a security goal. This has create a specific market (Siemens, Schneider, ABB ...)

Following the PC emerging market, the **IPC** (Industrial Personnel Computer) came later on the industrial field. It is based on a standard computer (software) but with a more reliable hardware.

Nowadays, the market propose the **PAC** (Programmable Automation Controller) or the **EPC** (Embedded Programmable Controller) both are computer running usually Window CE with a Real time application offering the functionality of a standardized PLC.

We have also the **EPS** (Embedded Programmable System) developed around FPGA or Risk Integrated circuits.

CoDeSys, can provide the binary for all of those implementations.
**INDUSTRIAL PRODUCT (SCHNEIDER).**

**SoMachine V4.x:** Software environment for the entire Machine

- It is a development platform dedicated for some of their products (M251, M258, ...).
- SoMachine is aimed at machine builder.
- It offers in a single environment all functions required by a manufacturer integrator.

SoMachine has a good integration with the Schneider HMI Vijeo designer through a common database.

Compared to Unity, SoMachine is a new (young) product built in a new software architecture, around the FDT (Field Device Tool) Frame Application. It is fully adapted to incorporate easily any device configurations. ([http://www.fdtgroup.org/](http://www.fdtgroup.org/))

This platform is running on Windows PC.

As core PLC programming they have incorporated a popular framework called CoDeSys.

CoDeSys V3.x: (Controller Development System) 3S-SmartSoftwareSolution

CODESYS offers a powerful IEC 61131-3 development environment with fully integrated backend functions such as compilers, debuggers, simulation and trace.

CoDeSys platform is easily integrated in many product.
• The development tools is unique and common for all.
• The control runtime systems (binary code) can be adapted for many different hardware architecture.

The version V3.x is developed with new software technologies based on Object Oriented Programming (OOP) and it is not fully compatible with the version V2.x (It is like PL7 versus Unity).
Over 250 firms use it, Beckhoff, 3M, Phoenix, Wago, ABB ...

You even get, for free, an implementation for the Raspberry PI

CPC- SoMachine tutorial
**INDUSTRIAL PRODUCT (SIEMENS).**

**WinCC-OA V3.11:**
(Windows Control Center- Open Architecture)

This product is the reference SCADA system used at CERN.
This product was initiated by the ETM company under the name of PVSS (old logo)

Often, the term PVSS, WinCC or WinCC-OA are used for WinCC-OA.

Other SCADA are also developed for our framework.
  - LabVIEW is planned as possible low end SCADA.
  - EPICS is used by external teams but NOT at CERN.

Our framework does also provides the necessary libraries for the configuration of Siemens or Schneider touch panels.
UNICOS: (UNified Industrial Control System)
It is a CERN-made framework design to develop a unique approach for all industrial control applications.
It cover all the layers from the Supervision to the Control Layer.

UNICOS proposes a method to design and develop the control application which will run with many commercial off-the-shelf products (e.g. SCADA, IPC, PLCs).

http://unicos.web.cern.ch
**CERN PRODUCT**

**UCPC V1.8.0:** (UNICOS Continuous Process Control) is a package of UNICOS. It provides a set of objects libraries in both, control and supervision layers.

This include Siemens and Schneider PLCs libraries in the control layer (Unity, Step7, SoMachine), and SCADA libraries in the supervision layer.

Often the term CPC, or UNICOS are used when it should be UCPC

CERN PRODUCTS.

**UAB V1.2.6:** (Unicos Application Builder) Proposes a set of tools used by peoples doing control systems.

Examples

- **Rev. Eng.**: recreate the source information from running applications.
- **Olproc**: Produce a list of objects based on project conditional template.
- **UCPC**: Generate all files used to implement the PLC and SCADA code.

http://unicos.web.cern.ch/automatic-generation-tools-uab
HARDWARE PLC

M258
Programming/Remote Access: Ethernet
Firmware update: USB
Application RAM: 10.4 Mbytes
Save Flash: 128 Mbytes

M251
Programming/Remote Access: USB/Ethernet/SD card memory
Firmware update: USB/SD card
Application RAM: 10.4 Mbytes
Save Flash: 128 Mbytes

CPC- SoMachine tutorial
As PC, I use Windows 7, 64bits, on a dual core CPU @3.16 GHz
It has 8 Gb of memory and a SDD hard disk.
SoMachine is demanding on memory and disk access.

The source code:
Under DFS; Applications\Schneider\So Machine\V4.1
• Run the Launcher.exe (Run as administrator)

License: free for two weeks, after you have to register.
CERN has a site license you don’t have to contribute.
The license is obtained by sending a CERN email to Schneider.
We don’t have the key Schneider deliver it.

This being done, we have now to activate some options.
When you start from scratch, under UCPC
Two libraries have to be added

Under the tab Tools tree, you extend the Application.
   Double click Library manager
   Add library
      Company: System
      System/SysLibs (gives access to SysTimeRtc)

      Company: Schneider Electric
      Util/ToolBox_Advance

Popular Library; OSCAT NOT needed:
   (Open Source Community for Automation Technology)
Choice made by developer to be independent of external library

http://store.codesys.com/oscat-basic.html?___store=en&___from_store=default
Add **PLCOpenXML** Command:

UCPC use XML format and this format are by default not present. Let's add them under the tap “project”
Open Tools/Customize/Menu/Project

Select the place were you are going to insert.

- Add a separator
- Add Command
  - categories : Objects
  - select Import PLCOpenXML … OK
  - select Export PLCOpenXML … OK

Add a separator

As you can see you can personalize your environment.
HARDWARE PLC INSTALLATION.

The major access to the PLC is performed via an Ethernet connection.

The key element:
The PLC must be configured with a good IP.

For that there is a BootP or DHCP software mechanism that query to a server for the IP associated to the MAC address of the device. The mechanism send a MAC broadcast frame in destination to all systems. Only those being configured as server replies. On CERN site we have one centralized server that make the job.

It Seems simple

But:
Schneider also configures inside this MAC broadcast frame a Broadcast IP address (Windows/Linux/Mac don’t do it). Broadcast IP means: For any PC even if their are not a server, all internal IP applications have to read the frame!
For security reason, on large network those Broadcast IP are blocked in routers.
CERN does it.
HARDWARE PLC INSTALLATION.

Also:
On Schneider modern products, when this query fails, the software assigns to the device an IP number based on default parameters. Once assigned it remains for ever.

Conclusion:
The PLC get an IP that you don’t expect and that does not correspond to the range of IP allowed on his network segment.

The work around:
Find a access that does not use IP.
• For devices like the PRM (Profibus Remote Module) there is no other way.
• For devices like Premium PLC we have the, well know, USB link.
• For M258 We have the SoMachine auto-discovery.
• For M251 We have the SD card, USB access or the SoMachine auto-discovery.
SoMachine hardware discovery.

SoMachine wizard:

The Wizard detect all PLC and gives a way to connect to them independently of their IP. It uses mainly the broadcast IP. It is then mandatory to have the SoMachine platform (PC) and the PLC on the same subnet!

In the laboratory area.
- You can prepare the M251/8 with the good IP and install it later in the production area.

In production area
- On M251/8, you can, connect a Portable with a twisted UTP cable to the PLC.
  !! On TN network, you cannot connect a Portable to the net.
- On M251 you have also the preconfigure a SD card or an access via USB.
Once connected you can configure the PLC with a good IP
CONFIGURE IP

Eth1 is the connection to SCADA
Eth2 is for other fieldbus devices. (Special for TM251MESE)

Select Eth1 define fixed address.
137.138.25.95/255.255.0.0 (not the one proposed by netops)

Gateway. Set a dummy address (137.138.1.1) because the LM25x screw up (for the PC) the CERN Gateway with funny request.

Disable all options (Production does not need all)
Except
SoMachine protocol active
FTP server active
Set PC IP (not needed) But SoMachine does not like pseudo class D netmask 255.255.255.192

Select Project/Project setting /Communication setting
“DialUp via IP or NodeName” OK

Double click My Controller
Controller tap (you should see all Mxxx devices)
Non configured PLC gives 11.11.51.205 (example)
Select it, Right-click and select “Process communication”
This gives the current setting detected to reach the PLC.
example: Fixed/mydevice/11.11.51.205/255.0.0.0/11.11.25.205
You have to download the apply with the Eth1 config.
on low banner select Nodename.
on tap Online select login, etc ...
then after some period the IP will change
Online help, on “search” type: “NAT”
you get “controller selection\”, “discovery protocol”
you get IP address configuration”
The PLC is loaded with its IP. We want to connect to it via a SoMachine PC located somewhere on the CERN network.

We open the application (baseline UnCPC)
On Project/ ProjectSetting / CommunicationSetting/
    select dial up via “IP address”
On Devices tree double click “MyController” to get “the controller selection”.
On the bottom select

Ipaddress via NAT (remote TCP) Nat address 137.138.25.85
/1105/137.138.25.85 press test.
PLC should reply, now you have its NodeName
“TM25xxxx@xxx” and you can connect with
NodeName via NAT (remote TCP)
The Ip of the PLC/1105 TM25xxxx@xxx. Test.

• On Online tap select login ....
PREPARE THE BASELINE

From the package, open the baseline.
  Change the PLC according your product.
  Add all IO modules to have the image of the real application

Save it under a new name:

Do a backup of this save, usually you are going to overwrite it by mistake...
PREPARE THE APPLICATION

From the Functional analysis document and/or the design document you prepare the excel SPEC sheet.

You can prepare also some logic templates that will be associated to field object, or you leave the UCPC to create an empty template for each field object.

You run UCPC and you get files ready to be loaded into the baseline.

Open the baseline.
Load files
Run
That all

For the SCADA you have to load the database as usually. The communication protocol is TCP/IP Modbus.
SoMachine is a rather new product.

It take quite a lot of the control of your PC.
Don’t be surprise that you get a stuck PC for more than 30 seconds.

The platform can crash and you loose the work!
Save your work from time to time.

When it crash and if you save it, your saved version will be corrupted...

The best, save your work from time to time in different name, or do Ctrl C, Ctrl V to the saved version.
In case of corrupted version, try to delete all sections modified since the last build. It seems that the editor place some non asci characters (not visible) that corrupt everything.

For the SCADA, there is nothing special:
You have to load the database as usually.
The communication protocol is TCP/IP Modbus.
Maybe you will have to configure the IP redirectoring.
POSSIBLE IMPROVEMENT

- For M251 and M258 the processor resource is shared with the CoDeSys application and the TCP/IP driver. (This is not the case on High End IPC available on the market).
  The resource is then limited. Further more the TCP/IP concurrent open sockets is limited to 8 (On premium is may goes above 20).
  The TSPP is designed for an architecture expecting 20 sockets, and in M25x it may produce problem if you are implementing a big project.
  A review of TSPP for this type of product could be envisaged.

- The direct addressing should be avoided. It means that UCPC instances have to be review, the TSPP should be reworked using logic addressing maybe via data array. (this will save a large memory space)

- With the coming disponibility of OPC UA server inside PLC, a new TSPP mechanism should be proposed.
  With this solution, there is no more communication middleware in the PLC, and the Scada will have also a light specific implementation.