This course was designed to meet the “Specification of requirements for a standard IEC 61131-3 training course” as presented in the file PC2TR_10_.doc available from the PLCopen.org web site.

IEC 61131-3 Overview (required for PLCopen compliance)

Using PowerPoint file from PLCopen.org

IEC 61131-3 CoDeSys implementation of the standard

The IEC software Model

Configurations Resources Tasks
Global variables
Access paths

Elements

Character set
Rules used to build identifiers
Language keywords, comments

Variables

IEC Types User defined types
Global vs. Local
Persistent variables Retain variables

Functions

Variable declarations
Function Body
IEC Standard functions User defined functions

Function Blocks

Variable declarations
Function Block Body
IEC Standard function blocks User defined function blocks

Concluding Slides

Textual Languages

IL
- Common elements
- Instruction set
- Operators, modifiers and operands
- Calling Functions and Function Blocks

ST
- Common elements
- Instruction set
- Operators, modifiers and operands
- Calling Functions and Function Blocks

Concluding Slides

IEC 61131-3 programming with the SMLC

Demo the class application

Basic features of the CoDeSys IDE (required in order to do the labs)

Creating a new program – target settings
Editing a program
Compiling a program

Overall program structure

The advantages of SFC
State machines

Sequential Function charts

Common elements
- Instruction set
- Operators, modifiers and operands
- CoDeSys “simplified” steps (vs. IEC steps)

Exercise 1 First look at CoDeSys;
Create SFC for 2 axis CutToLength

Continuting to develop your program

Exercise 2 Create enumerated "STATE" data - Create transitions in ST

I/O Configuration

Exercise 3 Configure I/O

Graphical Languages

LD
- Common elements Instruction set
- Operators, modifiers and operands
- Calling Functions and Function Blocks

FBD
- Common elements Instruction set
- Operators, modifiers and operands
- Calling Functions and Function Blocks

ServoWire Pro

ServoWire Pro is a configuration and commissioning utility for use with the ORMEC SMLC [ServoWire Motion and Logic Controller] and ORMEC ServoWire Drives.

It is the SMLC which runs the IEC 61131-3 compliant code. The drives communicate with the SMLC using ServoWire [IEEE-1394] cables, and provide the servomotor control.

ServoWire Pro provides a means to configure the Ethernet ports on an SMLC, to send / retrieve setup and other files with the SMLC, to upgrade firmware on the SMLC, and in addition provides drive configuration utilities.

SwSetup - Setup files

- Project settings
- Drive selection Drive settings
- Motor selection Custom motor editor Other motor Settings
- User units

SwPro communications setup

- Ethernet
- Serial – creating a dial-up networking connection
- CoDeSys communications setup

SwMonitor

SwUpgrade

SwTune

SMLC Utilities

- SMLC Sync
- SMLC Upgrade
- SMLC Ethernet configuration
- SMLC File utility

Exercise 4 Running SwMonitor, SwUpgrade, SwTune, and the SMLC utilities

The SMLC Function Block library

Standard Libraries

Using the Library Manager
- Review existing libraries [6]
- Add more? Where do they go?
- ORMEC now is supplying three additional libraries.
- ORMEC Utility library.
- ORMEC drive library.
- ORMEC miscellaneous library

PLCopen basics

Enable vs. Execute
Events

PLCopen MC_ function block overview

Review the PLCopen function blocks that are implemented in the SMLC

ORMEC MC_ function block overview

Review the ORMEC “vendor specific” function blocks

Continuting to develop your program

Opening axes

Exercise 5 Create Init using LD

Debugging aids

Browser commands (smlclog, smlcstate, smlcaxis)

Exercise 6 Create Estop, & Manual Mode [Jog both axes] using LD

Project Options

Visualizations

Exercise 7 Start visualization [EStop, Reset, Jog]

Working with PLC_PRG - Task Configuration

Timing Guidelines
Sampling Trace
Watch Window

Exercise 8 Tasks

Exercise 9 Open SMLC_Example and review

Project Compare

IEC 61131-3 Training Course Syllabus - Day 3

IEC 61131-3 programming with the SMLC (continued)

Exercise 10 Sampling Trace and Watch Window

Error Handling

Exercise 11 Add Error Display to HMI Visualization
Add ability to change feed speed
Add display of current state

Project Document

Boot projects – target setting
Choosing a boot project
Storing program source on SMLC
Password protection

Other Topics

Useful CoDeSys configuration options
OPC Configuration
Review the PDF from CoDeSys located on the 3S web site.

Creating reusable function blocks
In SFC (list advantages for sequential operations, homing example)
SFCInit, SFCReset, SFCCurrentStep
In ST
In LD

Exercise 12 Creating a reusable function block

Advanced features
PLSs
Exercise 13 PLS
Cams / Profiles
Exercise 14 Cams

HMI
HMI Configuration

Creating libraries / licensing
Internal libraries:
These libraries are created in CoDeSys, provide additional functionality, and the source of may be hidden.

Users and OEMs can create their own libraries, and license to others, if desired.
On install CoDeSys asks for a license if the license feature was implemented by the User or OEM.
External libraries:
These libraries are written in C++ or a similar language, and are only written by a CoDeSys OEM [such as ORMEC]. To create these libraries requires purchasing the documentation and license from 3S.

The SMLC library is an example of an external library, and was written in C++.
ORMEC does not require a license number for any of its CoDeSys libraries.