ISEG HV Modules for STRAWS
(Server: SLiC OPC / Client: PVSS Panels)

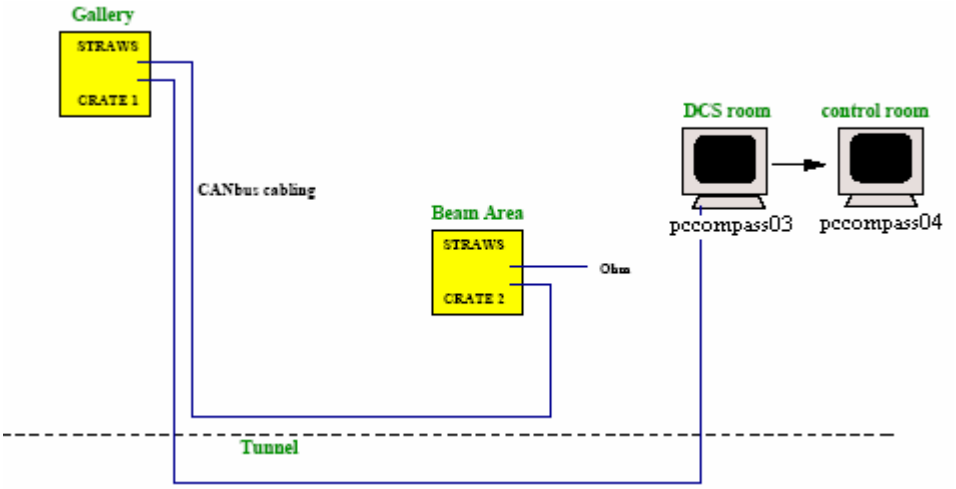


Figure 1. Cabling for HV STRAWS Crates/ DCS control room

1. Hardware: Crates/HV Modules

There are two crates for the Straws HV modules. The first one is called ST03-ST04 and it is located in the Jura side. There are, in this crate, six ISEG HV modules: five connected to the chambers and one used as spare. All the six modules have the 3.15 Firmware version.
The second crate is called ST05-ST06 and it is located in the Saleve side. There is, in this crate, four HV modules: three connected to the chambers and one used as spare. All the four modules have the 3.15 Firmware version.

2. SLiC OPC Server developed in C++

SLiC (Slow Control framework) is an extensible, configurable software framework for device control on front-end computers, intended for use when commercial front-end software does not provide the functionality or device access required. SLiC facilitates the integration of devices into a DCS through the reuse of previously developed device driver code and the easy integration of device driver code for devices not currently held in the SLiC library.

A new driver was developed to control the ISEG Straws HV modules. Following the SLiC structure, it was written in C++:
The main project was developed in *pcitco20,* which has a special license to build OPC servers. This machine belongs to ITCO group and the server can **only** be developed or modified in it.

SLiC framework was installed on *pcitco20* in August/2003.

2.1. SLiC development environment set-up for Windows NT/2000:

The following details how to set up the development environment for SLiC under Windows NT. The procedure assumes that VC++ v 6 is being used.

Installing Dependent Libraries

SLiC depends on the following libraries:

- Softing OPC for OPC communication access;
- DIM for DIM communications access;
- STLPort for platform consistent thread safe STL containers;
- ACE for platform independent threading, IPC etc;
- Xerces for xml parsing of the configuration file.

Create Directory <SLIC_DEPS_BASE> and set the environment variable SLIC_DEPS_BASE to that directory name.
Make directory <SLIC_DEPS_BASE>\DLLS. Set the PATH environment variable to include this directory.

These libraries are to be installed as follows:

Dim_v14r2.zip (http://dim.web.cern.ch/dim/dim_wnt.html) – extract in <SLIC_DEPS_BASE>. In **tokenstring.hxx, sllist.hxx, dic.hxx & dis.hxx** in directory <SLIC_DEPS_BASE>\DIM\dim change the include file name <iostream.h> to <iostream> (use new style iostreams instead of the old ones). Copy <SLIC_DEPS_BASE>\dim\bin\dimStd.dll to <SLIC_DEPS_BASE>\DLLS.

STLport-4_5.zip (http://www.stlport.org/download.html) – extract in <SLIC_DEPS_BASE>. There is no need to compile as we are not using their iostreams. However you MUST make sure the following macros are uncommented:

```
#define _STLP_NO_OWN_IOSTREAMS 1
#define _STLP_DONT_REDEFINE_STD 1
#define _STLP_WHOLE_NATIVE_STD
```

and

```
#define _STLP_USE_OWN_NAMESPACE 1
#define _REENTRANT 1
```

are defined in the file <SLIC_DEPS_BASE>\STLport-4.5\stlport\stl_user_config.h

Ace-5_2.zip (http://gd.tuwien.ac.at/infosys/prot/ACE/) – extract in <SLIC_DEPS_BASE>. Create a file called config.h in the <SLIC_DEPS_BASE>\ACE_wrappers\ace directory that contains:

#define ACE_HAS_WINNT4 0
#define ACE_HAS_STANDARD_CPP_LIBRARY 1
#define ACE_HAS_MFC 1
#include "ace/config-win32.h"

Now load up the project file for ACE
(<SLIC_DEPS_BASE>\ACE_wrappers\ace\ace.dsw).

Set Project Configuration to ACE.DLL-win32 debug (default setting) and build.
Copy the DLL <SLIC_DEPS_BASE>\ACE_wrappers\bin\aced.dll into
<SLIC_DEPS_BASE>\DLLS
xerces-c2_1_0-win32 (http://xml.apache.org/xerces-c/index.html) – extract in <SLIC_DEPS_BASE>. Copy <SLIC_DEPS_BASE>\xerces-c2_1_0-win32\bin\xerces-c2_1_02D.dll to <SLIC_DEPS_BASE>\DLLS

Installing SLiC

Extract slic.zip in <SLIC_DEPS_BASE>. Goto
<SLIC_DEPS_BASE>\SLiC\src\WinSLiC and start the project (winSLiC).

Make sure that the following environment variables are defined to the correct directory:

SLIC_SRC_BASE (<SLIC_DEPS_BASE>\SLiC\src)
ACE_BASE (<SLIC_DEPS_BASE>\ACE_wrappers)
Path (add in the end: ; <SLIC_DEPS_BASE>\DLLS)
STL_BASE (<SLIC_DEPS_BASE>\STLport-4_5\stlport)
XML_INCLUDE_BASE(<SLIC_DEPS_BASE>\xerces-c2-1_0-win32)

2.2. SLiC OPC Server development:

The header file is called **IsegHVPS_EHQ.h** and the .cc file is called **IsegHVPS_EHQ.cc**.
The files are located in <SLIC_DEPS_BASE>\SLiC\src\libcommondevices directory.

The binary file is called **SLiCApp.exe** and the config files are located in <SLIC_DEPS_BASE>\SLiC\src\winSLiC\SLiCApp\Debug directory.
The config files needed are **config.dtd** (generic config file needed for all slic servers) and for Straws: **configIseg.xml.** In this last one, HV modules are defined to be accessed by the SLiC Server (including the ID addresses for each module) and read by the PVSS Client.

2.3. SLiC OPC Server application:

A directory was created on the COMPASS windows machine *pccompass03* (C:\SLiC_Iseg). In order to run the SLiC Server the following files were copied there (from *pcitco20*):

- All the dll files from the <SLIC_DEPS_BASE>\SLiC\src\winSLiC\SLiCApp\Debug directory.
- Config.dtd
- Config.xml
- configIseg.xml
- SLiCApp.exe

In order to execute slicapp.exe automatically associated to the correct config file (configIseg.xml) an environment variable was created:

Variable: SLIC_STSIL_CONFIGFILE
Value: C:\SLiC_Iseg\configIseg.xml

This way, every time the PVSS OPC manager for Straws starts, slicapp.exe will also start automatically, linked to configIseg.xml file.

Before running Slicapp.exe for the first time, a registration is needed.
To register SLiC in the computer where it is intended to be running you need to do the following:

- Open a dos window;
- Move to the directory where the slicapp.exe is located (C:\SLiC_Iseg);
- Run the command line: slicapp –register

3. PVSS Client: client configuration and panels scheme

3.1. Client configuration

The panels for ISEG HV Straws were developed on the linux machine: *pccompass04*.
In the PVSS config file (pccompass03) the following two lines were added:
[opc_12]
server = "SLiCStrawSilicon" "CERN.STRAWSILICON_SLIC_OPC"

This means that the OPC Manager, in PVSS, with number 12 will receive items from a server called *CERN.STRAWSILICON_SLIC_OPC* and send it to PVSS datapoints by the OPC driver created in PVSS named *SLiCStrawSilicon (server name in PVSS)*.

Because SLiC Server is sending ≈ 5000 items each cycle, different OPC groups were created on the PVSS compass_opc project:

Straw_status123group → contains the items *getCurrent, iLimitStatus* and *vLimitStatus* for each module. These items are used to reset automatically the alarms, just by reading them. Because the reading of these items needs to be slower than the others, the update rate is 10 seconds. For the others the update rate time is 2.5 seconds.

Straw_group_i0 → contains the items for all channels in module0

… ….
… ….

Straw_group_i10 → contains the items for all channels in module10

Straw_group → contains the group items for all channels/modules, except the ones mentioned in Straw_status123group.

3.1.1. Datapoint structure

For ISEG HV Straws, a datapoint type was created under dcs1, named isegHVM. The names for the datapoints start by "Module" followed by the module number. Example:

+isegHVM
+Module1
+Channel0
<Properties>
…
+Channel31
….
+Module9

This way, the Voltage monitored value for the Channel0 of the Module1 could be found in: "dcs1:Module1.Channel0.VMon"

3.2. Panels scheme

In the PVSS main panel a button for straws can be found (Figure 2).

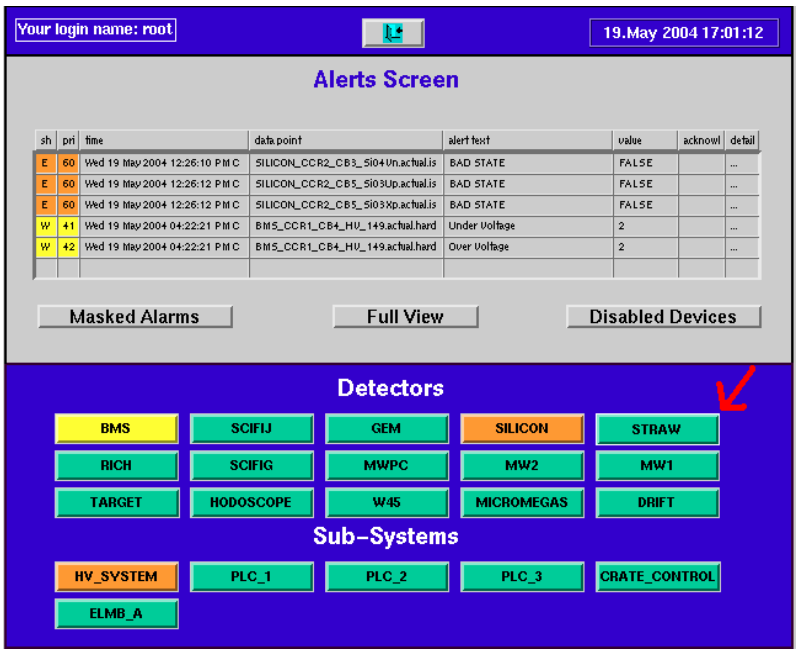


Figure 2. PVSS Main Panel for DCS

This button gives access to the Straws HV Panel (figure 3).

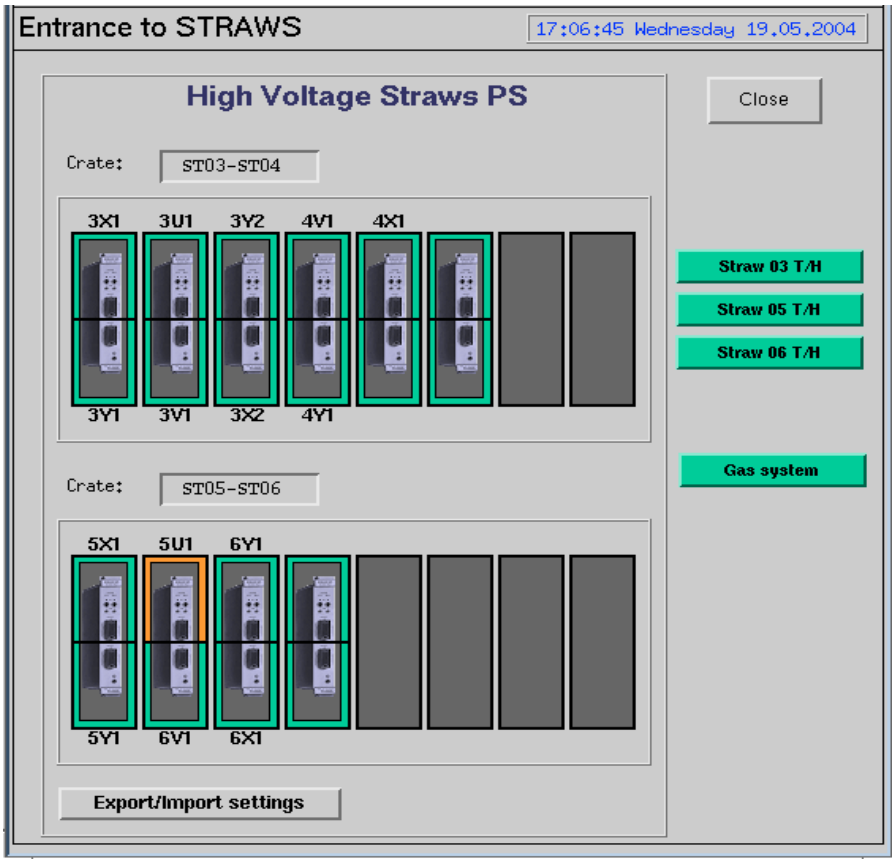


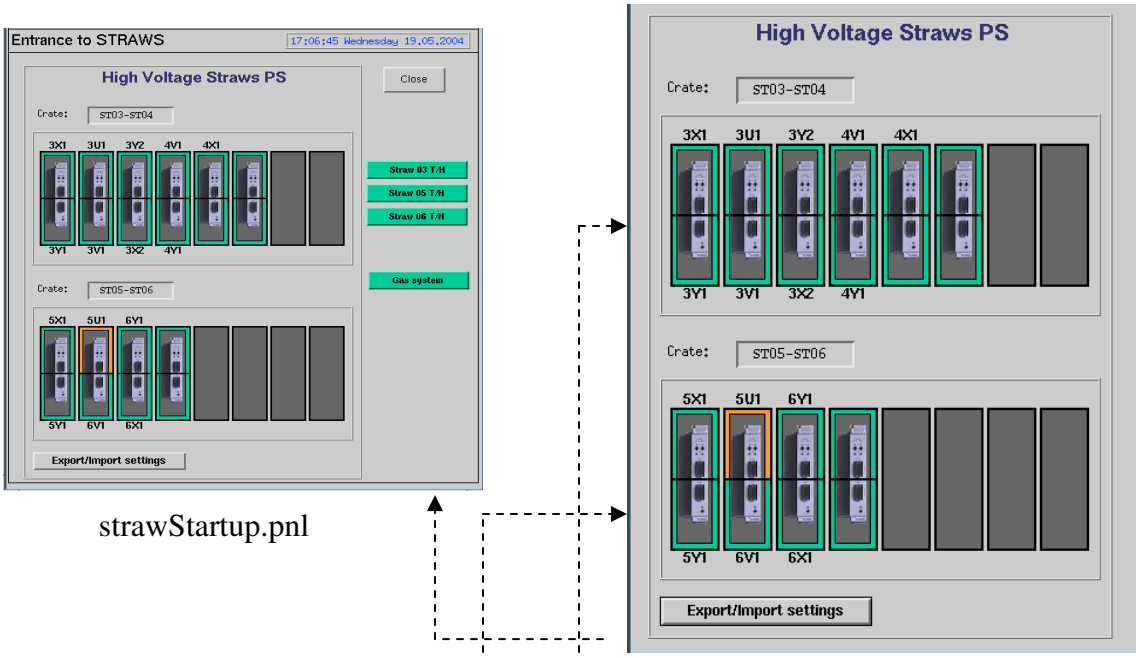
Figure 3. Straws HV Panel (strawStartup.pnl)

In this panel (Figure 3) each module powers two chambers. The chamber names are explicit in the panel. By clicking in a "sub-module" (chamber) the user opens a "sub-module" panel which contains the values/settings for the 16 corresponding channels

3.3. Panels scheme: development

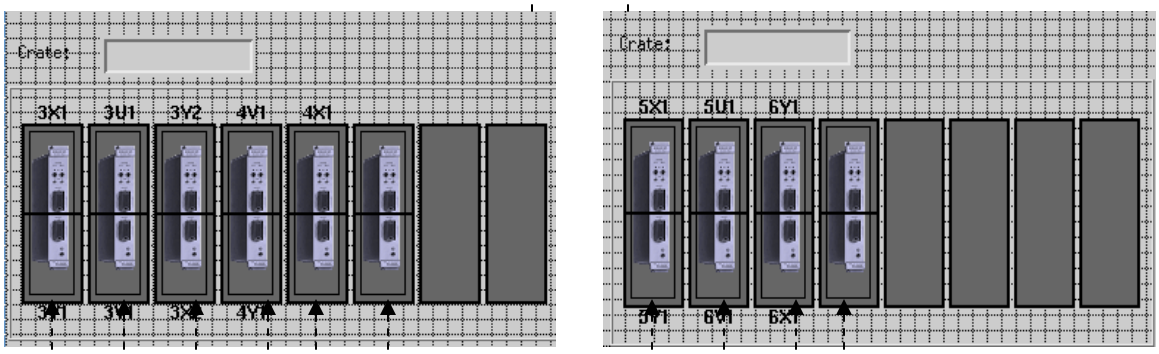
Linked to reference panel

Linked to panel



strawStartup.pnl

IsegCrate.pnl



IsegCrate_1a.pnl

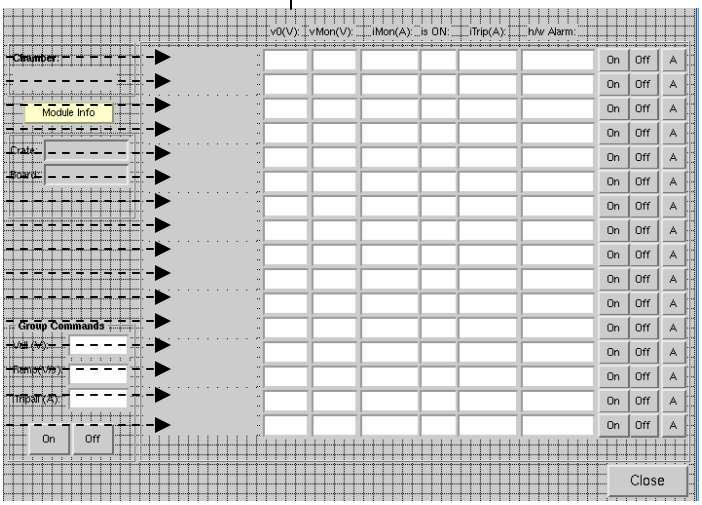
IsegCrate_2a.pnl

IsegModule1.pnl

By clicking on IsegModule1.pnl panel, depending on the chamber chosen, two different panels can be displayed: Module_channels1.pnl (refers to the firsts 16 channels: sub-module in the top) or Module_channels2.pnl (refers to the lasts 16 channels: sub-module in the bottom).

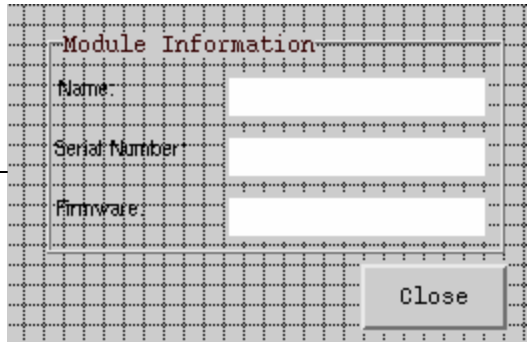
IsegModule1.pnl

Module_channels1.pnl

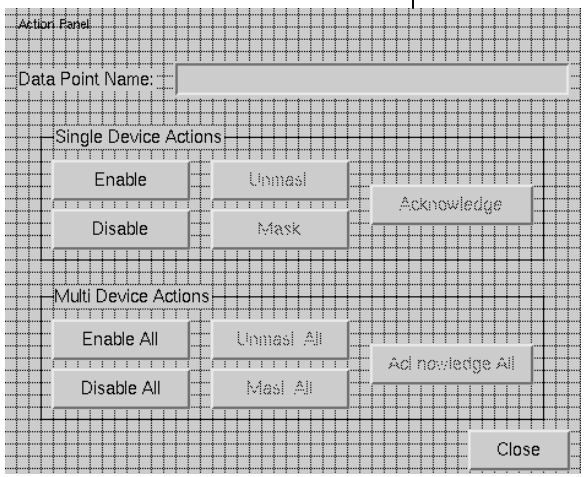


Module_channels2.pnl

channel_values1.pnl

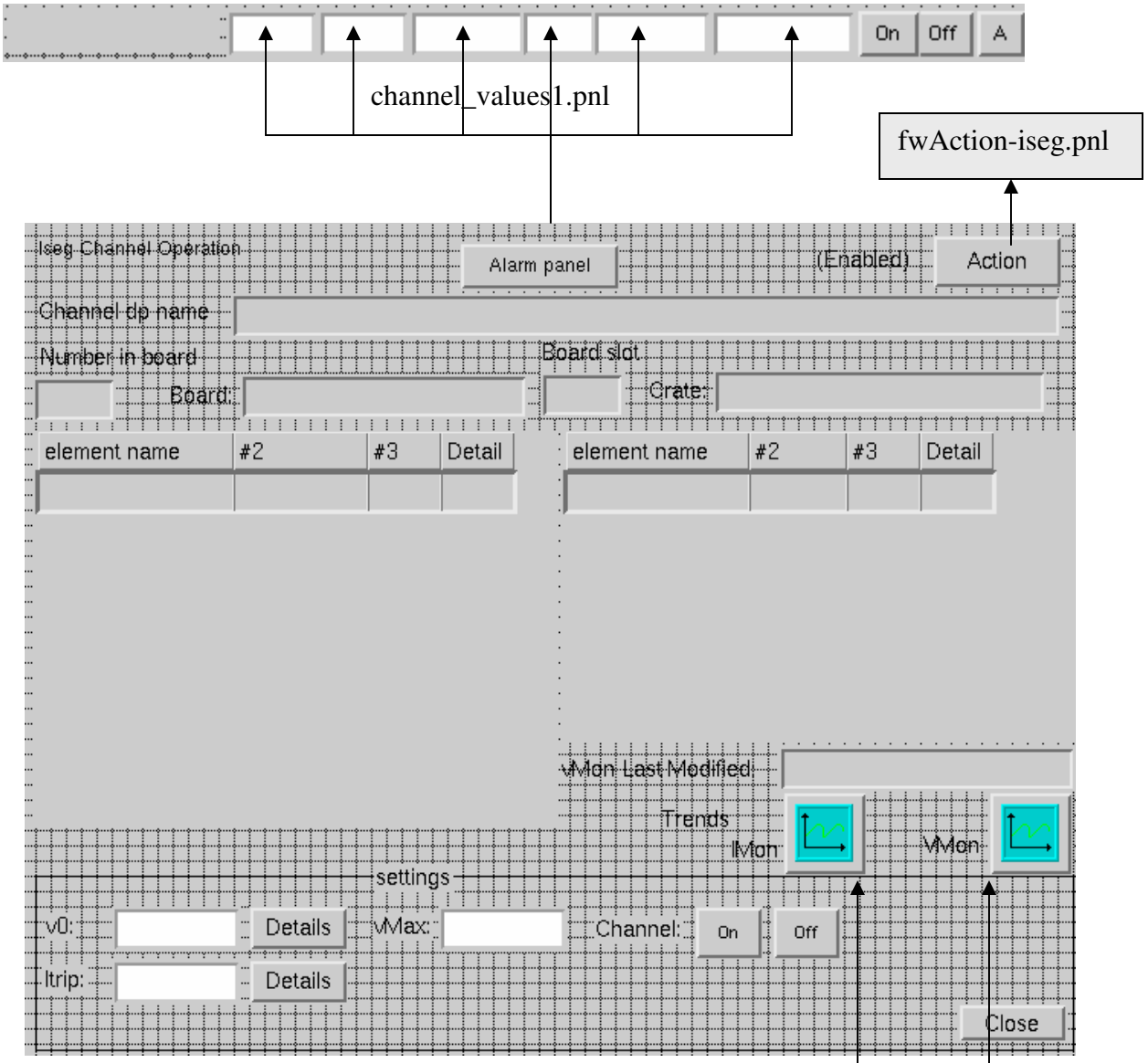


ModuleInfo.pnl

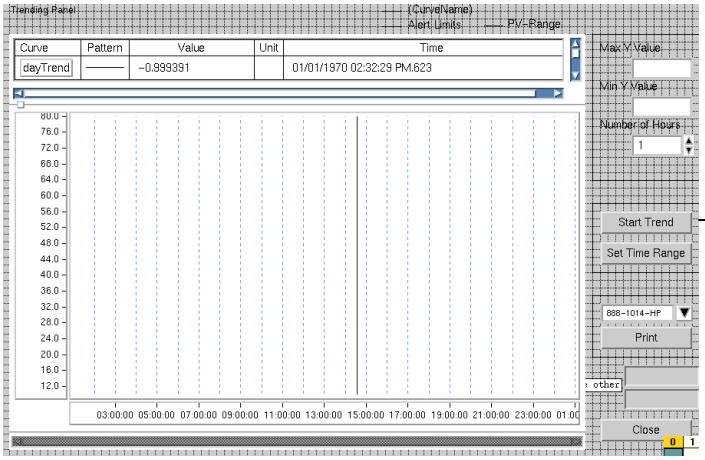


fwAction-iseg.pnl

In one of the reference-panel (channel_values1.pnl) of the sub-module panel (i.e. Module_channels1.pnl), by double-clicking in one of the fields (ex: vMon) the IsegChannelOperation.pnl will be displayed for it, as showed in the next picture.



IsegChannelOperation.pnl



fwTrendPanel_f.pnl

To start the OPC Client/Server for HV Straws, the PVSS OPC manager 12, in *pccompass03,* needs to be started. It will automatically run SLiC server for Straws and after one or two minutes the control system for HV Straws is ready.

David Sora, 20/05/2004