saia-burgess

Control Systems and Components

DDC Suite 2.0 / Version 2.1 PG5 2.0

Table of Content

PG5 Building Advanced / DDC Suite 2.0
Overview developing DDC Suite

2000	**Starting developing first FBoxes**
…	**… in lot of projects tested and always improved …**
2004/April	**FBoxes attained the core style and functionality**
2004/November	**DDC Suite became SBC product - version 1.0 Germany/Netherland**
2005/March	**Fupla editor adoption to improve mechanism – version 1.3**
2006	**Annual update with improved functionality – version 1.3.x**
2007	**Annual update with improved functionality – version 1.3.y**
2008/June	**Annual update with dramatic development – version 2.0**
	- HDLog implemented
	- Alarming implemented
	- BACnet implemented
	PG5 license management

General basics

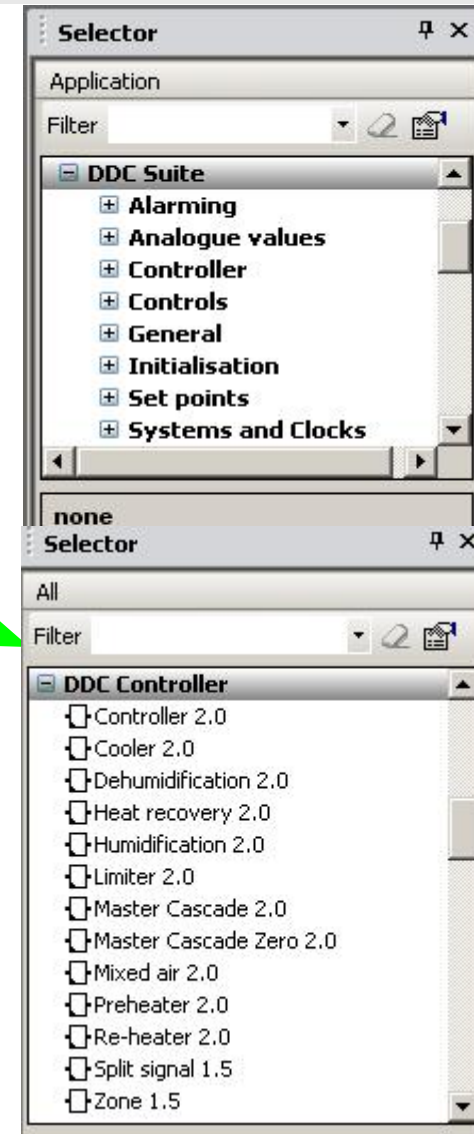# Overview

PG5 Building Advanced / DDC Suite 2.0
General

The base of DDC Suite is a FBox library containing 8 FBox families.

This families are representing the FBox functionalities and features like alarm management, controlling, sensor monitoring and so on.

As you can see in FBox family DDC Controller the FBoxes have been designed and dedicated for Heavac (Heating, Air Conditioning, Ventilation, Cooling) applications – but they can of course also be used for other systems.

But in PG5 package there is already a Heavac FBox library with some FBox families available. What's the difference between Heavac and DDC Suite library?

Selector

Application

Filter

- DDC Suite
  - Alarming
  - Analogue values
  - Controller
  - Controls
  - General
  - Initialisation
  - Set points
  - Systems and Clocks

Selector

All

Filter

- DDC Controller
  - Controller 2.0
  - Cooler 2.0
  - Dehumidification 2.0
  - Heat recovery 2.0
  - Humidification 2.0
  - Limiter 2.0
  - Master Cascade 2.0
  - Master Cascade Zero 2.0
  - Mixed air 2.0
  - Preheater 2.0
  - Re-heater 2.0
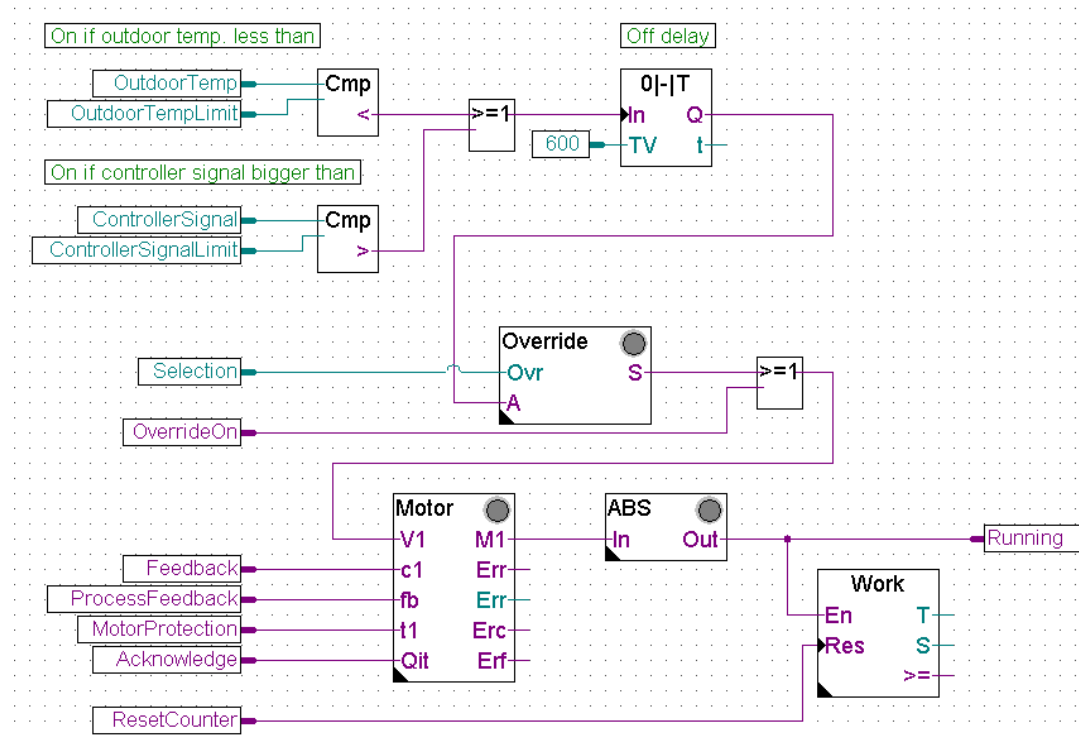 - Split signal 1.5
  - Zone 1.5

PG5 Building Advanced / DDC Suite 2.0
PG5 Building Standard (Heavac)

Let's have a look at a typical pump for a air heater. This pump should have at least this standard functionalities:

- switch on if outdoor temperature is less than x °C (e.g. forward-thinking frost protection)

- switch on if controller signal valve is greater than x %

- manual override e.g. for maintenance or commissioning

- switch on if forced e.g. frost protection

- counting working hours and feedback
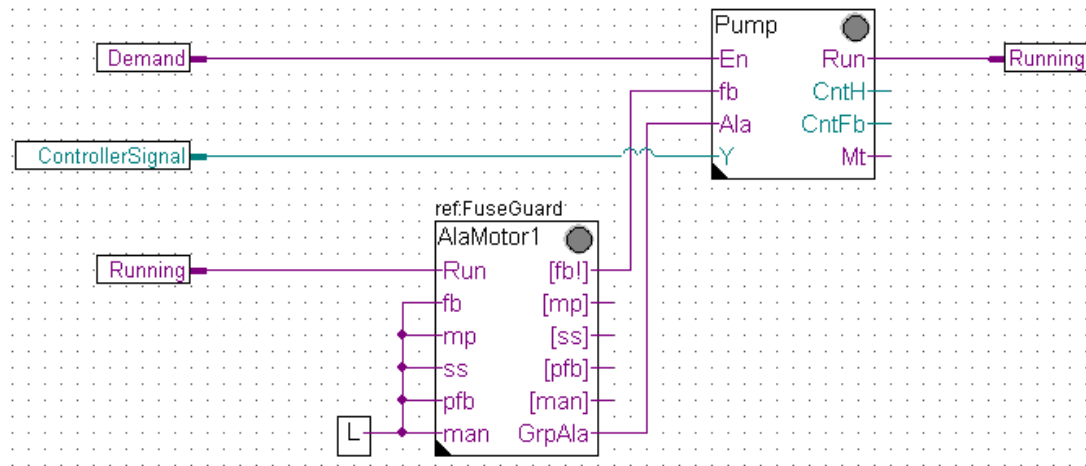
- Anti blocking protection

All this can be programmed with Heavac FBox library and may look like this example

The same pump functionality programmed with DDC Suite looks much easier:

- all functionalities you have to program by yourself (e.g. frost protection, manual override, counting working hours …) are already implemented in the FBox

- the FBoxes are much more complex. Standard functionalities you have to think about (and first you have to know them – what makes sense …) are included

- and all this functionalities and features can be activated or changed in the FBox – online, without reprogramming, compiling and downloading. E.g. if you have to turn off the anti blocking function from a pump – just disable it online!

The first difference:

Higher integrated FBoxes. Know how and experiences from systems engineering has been moved into the library, families and FBoxes

If we compare both fupla pages (Heavac and DDC Suite) we can find additional advantages:

- reading and understanding the Fupla is easier – less FBoxes on 1 page, less connections
- clear and well arranged – easier to handle for e.g. a new member in programming or service team
- easy to maintain

This is not the single difference – but this is the first impression.

Let's have a look at the data of this pump control.

PG5 Building Advanced / DDC Suite 2.0

PG5 Building Standard (Heavac)

Data for pump control programmed with Heavac must be defined by the SI himself. Only this data are available in symbol editor – no further information which FBox has been used ..

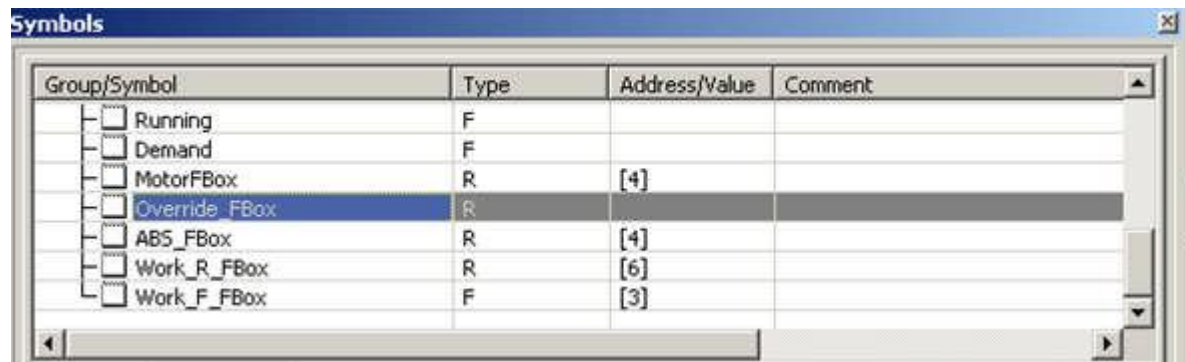Each FBox contains more or less data and they can get a symbolic definition in detailed adjust window to access them in symbol editor. Sometimes a FBox needs more than 1 symbolic definition (e.g. 1 for register and 1 for flag).

Per FBox at least 1 symbol must be defined – and this symbolic data contains more information, arranged in an array.

You can find some information in detail windows or in help file – but in this moment you can't use the symbols with sense.

If you have to give the data points to a SCADA system engineer (also Web or HMI) – he's not able to do anything with it. Too less information.
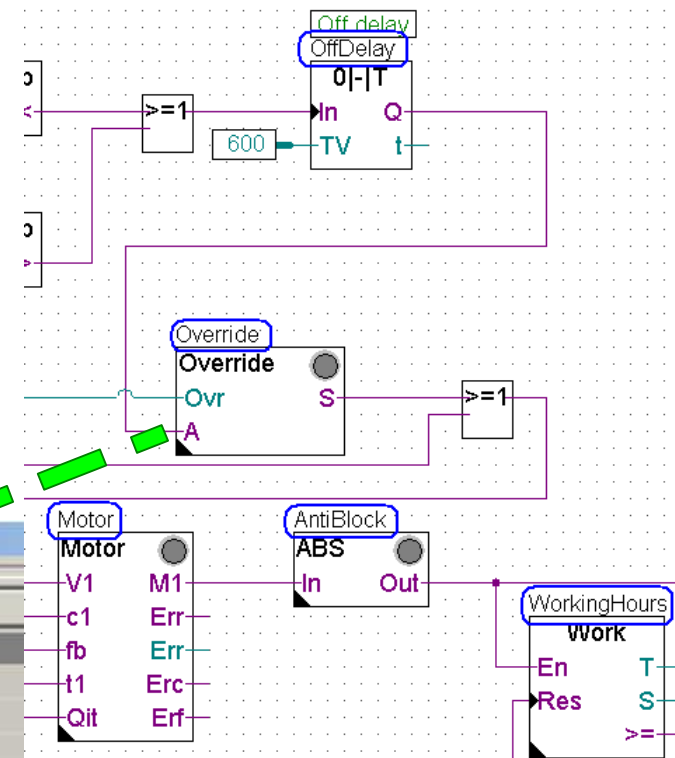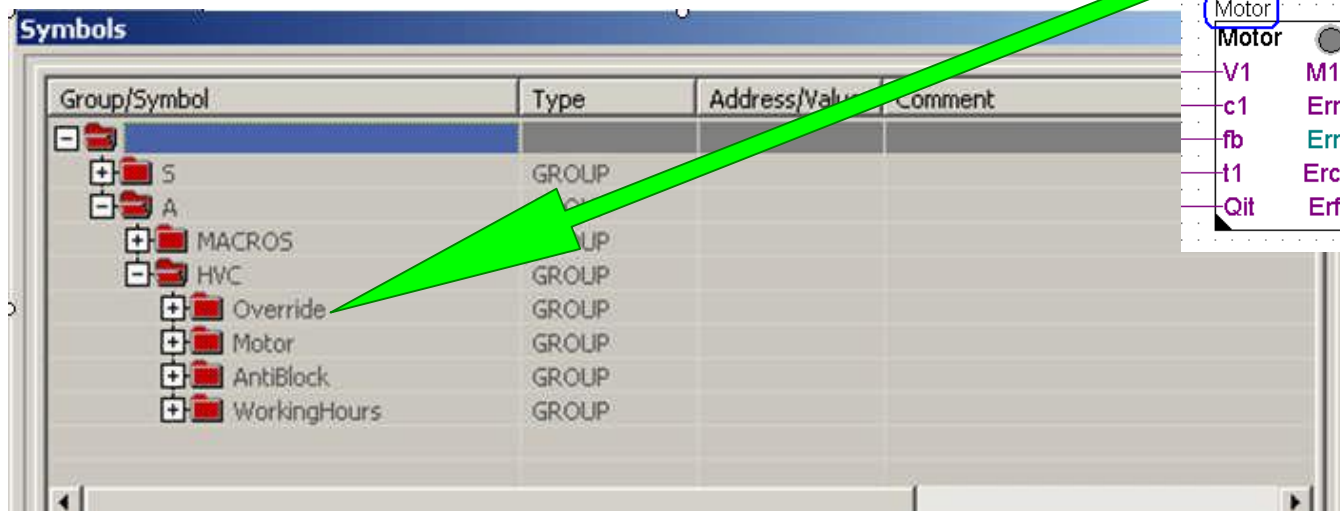
PG5 Building Advanced / DDC Suite 2.0
PG5 Building Standard (Heavac)

There is an additional possibility to get detailed symbols and a little bit more information – half automatically. Therefore each FBox must get a text in FBox property Name.

In symbol editor nothing happens – the program must be „build". After build in symbol editor a new tab System is available and therein the default group structure „A.HVC." is visible. Each FBox will create a subfolder using the text from FBox property name. But this will only work if the build has been successfully.

Each subfolder contains detailed information from the FBox, some are still marked as array – []-brackets – but no symbol supporting further information. E.g. whats the meaning of FeedBackT or FeedBackD?

And the symbols are only available after a successful build – they are not immediately available when the FBox is set into Fupla page or FBox property name is set. Some FBoxes do not support this mechanism, at least only Heavac FBoxes – excluding the family Test.

PG5 Building Advanced / DDC Suite 2.0
PG5 Building Advanced (DDC Suite)

Pump data with DDC Suite are created fully automatically – just put the FBox into Fupla page!

DDC Suite FBoxes are not using arrays – always separate data points. The comment of each data point should explain the functionality and already prepared an indicator for SACAD systems to optimize the communication load (advice)

Simply by adding a FBox in Fupla page all symbols for each data point are available – immediately! The are generated in groups, the first group indicates the family, the second the FBox itself.

The symbols itself should not be renamed, changed or deleted – they are part of a unique data base name convention!

But the groups can be renamed or moved into other groups and maybe structured by location, systems or what ever.

Group/Symbol	Type	Addre...	Comment
Alarming	GROUP		
Motor1_0	GROUP		
BrmDI	R		(5) Digital Input feedback
BrmSm	F		(1) Alarm feedback missing
BrmVerzoeger	R		(5) Maximum delay until feedback operation is presen
DrzDI	R		(5) Digital Input process feedback
DrzNoNc	F		(5) Selection of the normal status of the process feed
DrzSm	F		(1) Alarm process feedback
DrzVerzoeger	R		(5) Maximum delay until process feedback is present
HandDI	R		(5) Digital Input manual override switch
HandNoNc	F		(5) Selection of the normal status of the manual switc
HandSm	F		(1) Alarm manual override active
HandSpgGrp	R		(5) Associated voltage group for suppressing ghost al
MotDI	R		(5) Digital Input motor protection
MotNoNc	F		(5) Selection of the normal status of the motor protec
MotQuitPflicht	F		(5) Selection whether the alarm follows the input or re
MotSm	F		(1) Alarm motor protection
MotSpgGrp	R		(5) Associated voltage group for suppressing ghost al
RepDI	R		(5) Digital Input service switch
RepNoNc	F		(5) Selection of the normal status of the service switc
RepQuitPflicht	F		(5) Selection whether the alarm follows the input or re
RepSm	F		(1) Alarm service switch off
RepSpgGrp	R		(5) Associated voltage group for suppressing ghost al
SsmTyp	R		(5) Selection of the group alarm
Controls	GROUP		
Pump_0	GROUP		
AbsErlaubt	R		(5) Mode antiblock protection function
AnsteuerDO	R		(5) Digital output pump
Ansteuerung	F		(2) Display requestet pump state
Ausgang	F		(2) Display if pump should run
BedAt	F		(3) Display demand for outside temperature function
BedAtFunk	R		(5) Mode Function of Outside air temperature
BedAtGw	R		(5) Limit for outside air temperature
BedY	F		(3) Display demand for control signal function
BedYFunk	R		(5) Mode function of control signal
BedYGw	R		(5) Limit value of the control signal
Betrieb	F		(2) Corresponds to the input fb ≈ feedback
HMI	R		(4) Mode HMI lower priority
HMISuper	R		(4) Mode HMI higher priority
Nachlauf	R		(5) Turn off delay
Schaltung	R		(3) Number of feedback on
SchaltungMax	R		(4) Number of feedback on until message maintenanc
Sperre	F		(1) Motor blocked due to alarm
Stunden	R		(3) Number of operating hours
StundenMax	R		(4) Number of operating hours until message mainten

PG5 Building Advanced / DDC Suite 2.0
Summary 2

The second difference:

Fully automatically generated symbols when FBox is put on Fupla page

In contrast to half automatically created symbols from Heavac FBoxes (half automatically because you have to edit FBox property name and start a build) to DDC Suite FBoxes the created symbols

- are immediately available

- movable, restructured during engineering can

- are single data points with own symbols and comment

- format of each data point listed in help file

- suitable to build up a object oriented, component and/or system data structure

This 2 operative differences (more compact/complex FBoxes – fully automatically created symbolic data points) are representing the core of DDC Suite – easier, faster and better engineering.

DDC Suite 2.0 / PG5 Building Advanced
General overview

DDC Suite is an extension of PG5 containing

1. FBox library – the DDC Suite base. This FBoxes are higher implemented, using single data points and creating groups and symbols fully automatically

2. Fupla templates – predefined systems e.g. heating circuit, hot water, air condition – to start up in an easy way

3. Template objects for SWeb application – for each FBox graphical objects and adjust objects are available. Also for the predefined systems we have predefined Sweb system templates

4. Template objects in ViSi.Plus. During import data from Fupla into ViSi.Plus the FBoxes are detected and handled in ViSi.Plus data base again like FBoxes. At least not only the data points are imported – additional predefined alarm settings and historical trend information are generated automatically during import.

DDC Suite is not a totally different thing in PG5 - some FBoxes, Fupla and Web templates – and of course the FBoxes itself can be used without Sweb or ViSi.Plus. And they are compatible with Heavac FBoxes.

The target of DDC Suite is

Reducing engineering time – safe money - easier programming
Improve software quality – having higher "minimum standard level" then competitors

DDC Suite - Fupla

Please follow the teachers advice. Please

- use the same symbol names
- use the same group names
- place the FBoxes approx. at the same position
- do not work faster or different even if you are a "frequent PG5 user"

This workshop will show you some basic mechanism, structured workflow and well structured symbol organisation. Don't be afraid. You don't

- have to learn all FBoxes during this workshop
- have to be familiar with application programming
- must be a super programmer

If you just learn the mechanism and philosophy you'll understand the advantage SI can have with DDC Suite

DDC Suite 2.0 / PG5 Building Advanced
Syntax and remarks of actions during workshop

Click with left mouse button at this position

Double-click with left mouse button at this position

Click with right mouse button at this position

Follow the green arrow to next step

Example

Type in the blue text into the high lighted green text field

Watch this area

⚠ Changes/different workflow to former versions

Creating a new project

A new DDC Suite could be started within PG5 via „Project/New…".
But then the user has to integrate manually all he'd like to use, e.g.
HMI, BACnet and so on.

We recommend to start a new project via „Project/Restore…", that
means we using a predefined DDC Suite template project where a
lot of settings or pages, templates are already prepared.

Therefore those template projects include everything what can be used
in a project, e.g. Fupla template pages, SWeb template objects

Thereby a project backup will be a little bit larger (e.g. >5 MB) but
changes in future in this project may have no compatibility problem,
e.g. if in meantime a Sweb object template has totally changed.
Of course – our template project can be updated with your
templates/addendums and backup-ed as new template project.

So we are starting with „Project/Restore…" – by selecting a project template. Depending on template version or location we have to navigate to the correct folder …

You'll find project templates on PG5 CD in folder DDC Suite, get the newest templates from SBC Support HomePage www.sbc-support.ch within Software/PG5/DDC Suite.

When starting restoring the new project name can be defined. Please use

Workshop

Press "OK" and the template project will be used as base for our Workshop project.

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

If we have a look into our workshop project (use Windows File Explorer) we'll
see that already some folders are included:

- CPU-Template : a CPU template which should be used for each new CPE
we have to add in this project

- FbdToDms : Containing some information if a ViSi.Plus SCADA system
should be used

- And a PG5_Template containing :

  - FBox_AddOns : Containing some files for some language depending definitions
  and also some additional features
  - Fupla_Templates : here you'll find some predefined Fupla pages or systems, to
  be imported into Fupla
  - Sweb_Objects : Graphical objects and adjust objects (pages) for SWeb
  applications designed with SWebEditor.

```
Workshop
   CPU_Template
   FbdToDms
   PG5_Template
      FBox_AddsOns
         BACnet
         Documentation
         Konfiguration_IO
         SWeb_Alarming
      Fupla_Template
      SWeb_Template
```

Creating a new CPU in the project

First we have to create a new CPU – the "CPU_Template_xx" should stay as they are because therein some settings are already predefined.

Therefore we use the copy/paste in project manager. Click with right mouse button on "CPU _Template_EN" and then Copy in context menu.

Click with right mouse button on "Project 'Workshop'" and then Paste CPU in context menu.

We have to rename the CPU, please use "CPU001" and press "OK".

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Let's have a look into the new „CPU001" – here we'll see also
some predefined files:

- BACnet.bnt : User BACnet configuration file
- DDC_Alarming.CSV : automatically created text file,
  containing alarm texts for use within SWeb application
- DDC_BACnet.bnt : automatically created BACnet
  configuration file
- DDC_HDLog.txt : automatically created text file, containing
  detailed information how to record offline data
- HEAVAC.fup : prepared Fupla
- ReadMe.txt : Short description to the files described above

More details during the workshop!

We start engineering with a double click on file HVC.fup

NOTE: it is not possible to use DDC Suite Fboxes in more than
one FUPLA file per CPU.

Basic settings in Fupla

With DDC Suite we introduced some features in Fupla editor. Before starting engineering we have to check this new features – this is necessary only once time after installing PG5.

Select in Fupla Editor menu „View" and within menu list the entry "Options…". In dialog "Options" activate tab "Symbols" – here we see the new features for easier and better engineering within Fupla.

Use predefined Symbols:

When placing a FBox into a Fupla page a default group name space and symbols are created automatically in global symbol tab. **Recommended**

Display List before the symbols created:

Will pop up a dialog window when placing a FBox with predefined symbols to change/edit the default group name space immediately. If not activated the default group name space and symbols are used. **Recommended**

Navigate to the first static symbol when the Fbox selected:

Navigates in symbol editor to the first defined symbol used in the FBox you clicked on. Supports you to find the correspondent group/symbols in symbol editor. **Recommended**

Delete the internal symbols when the FBox removed:

If a FBox is deleted from Fupla page then also the correspondent group/symbol in symbol editor will be deleted. **Recommended**

I recommend to have new Fupla pages without side connectors.

Therefore please activate tab Layout and deactivate the checkbox New page with side connector.

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Remark:

Working with DDC Suite will force long structured tree structure in symbol editor.

Therefore I recommend to "undock" the symbol editor window and resize it to a bigger comfortable window.

You can enable or disable the symbol editor window with key "F5"

If you build up the program by placing some FBoxes you don't need the symbol editor window.

But for the first impression switch on the symbol editor window

At least we have to check a setting to be sure that the symbols are created in the right way.
Click with right mouse button into symbol editor, the select in context menu "Advanced" and in next context menu "Options …"

Please set ALL settings in dialog "Symbol Editor Options" to have the same behaviour!

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

First 2 pages containing the FBoxes always needed in a typically Heavac Fupla program.

- Init FBox from Heavac library
- BACnet device FBox to switch on/off BACnet feature for DDC Suite 2.0 FBoxes
- HDLog initialization FBox – general offline trending to be used in a SWeb application
- additional HDLog FBox for DDC Suite 2.0 FBoxes
- Alarm list FBox for a general alarm monitoring to used in a SWeb application

- AlarmHdr FBox - organisation alarm numbers for DDC Suite 2.0 FBoxes
- Init FBox from DDC Suite library (DDC Suite needs Heavac Init FBox placed in front of)
- Anti blocking FBox – central management of anti blocking method
- Fuse Guard FBox – (optional) monitoring fuses to hide "ghost alarms/messages" in FBoxes with alarm functionality.

In the Fupla you see in "Page Navigator" already a block with name "Systems". Double click on it to open the prepared empty page in it. You should have had this screen:

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

You should have this screen

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

We will create a very tiny air condition. The functionality should include some virtual functionalities:

- week clock to schedule the air condition
- system switch to start/stop the air condition
- a serialized start procedure to be sure that the air condition will start always in a proper way

And also some physical components:

- supply air temperature sensor
- exhaust air temperature sensor
- supply air fan, 1 speed
- exhaust air fan, 1 speed
- damper or valves for cooling,
  mixed air and heating, controlled

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

First Fupla page will contain

- the virtual functionalities week clock, system switch, serialized start procedure
- physical components supply air temperature sensor, exhaust air temperature sensor

1. Select from FBox selector tab Application the family DDC Systems and Clocks.

2. Click on FBox System 1 speed 2.0

3. Place FBox approx. at same position as you can see in picture (top/centered)

4. Immediately you should see that in symbol editor something has happened.

When placing a DDC Suite FBox in Fupla the FBox will generate automatically
- a main group like SystemClocks to indicate from which family this FBox has been selected
- a subgroup like System1_0 where the part before "_" indicates the FBox face name and after the "_" a index, starting from 0
- At least in this group all parameters from the FBox are available with symbol, type and comment

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Lets check the FBox adjust parameter. Double-click on FBox to open the Properties Window.

In the detail window you can see that the parameters are containing already predefined symbols and you can find these symbols again in the symbol editor.

**Using DDC Suite FBoxes will save a lot of manual work.
You never have to define symbols by your own!**

DDC Suite will do this for you automatically!

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

The FBox System 1 speed 2.0 gives us the possibility to start/stop e.g. the air condition

- Manually by selecting parameter HMI Lower priority

- optional by clock via FBox Input or SCADA

- optional by a calendar feature

Properties	
DDC Systems and Clocks:System 1 speed 2.0	
⊟ **General**	
(Name)	S01_Switch
Comment	
⊟ **Adjust Parameters**	
⊟ **System functions**	
BACnet	No
⊟ **Settings**	
HMI Low prio	Automatic
… Clock accessed by	Input
… calendar channel	Not used
… Requirement of clock	Off
⊟ **Static Symbols**	
… Clock accessed by	SystemClocks.System1_0.ClockMode F
… Requirement of clock	SystemClocks.System1_0.Clock F
Demand by operator	SystemClocks.System1_0.Demand F
Requirement through special	SystemClocks.System1_0.Special F
System mode	SystemClocks.System1_0.Required F
HMI Low prio	SystemClocks.System1_0.ModeLoPri R
HMI Higher prio	SystemClocks.System1_0.ModeHiPri R
… calendar channel	SystemClocks.System1_0.Channel R

DDC Suite FBoxes using always online parameters. Therefore it's possible to use e.g. during commissioning a clock FBox to start/stop the air condition as long the SCADA system is not present.

If the SCADA is online it's easy to use the SCADA clock manager to start/stop the air condition. Just switch the parameter … Clock accessed by from Input to SCADA. Now the SCADA can write the start/stop clock command into the parameter … Requirement of clock.

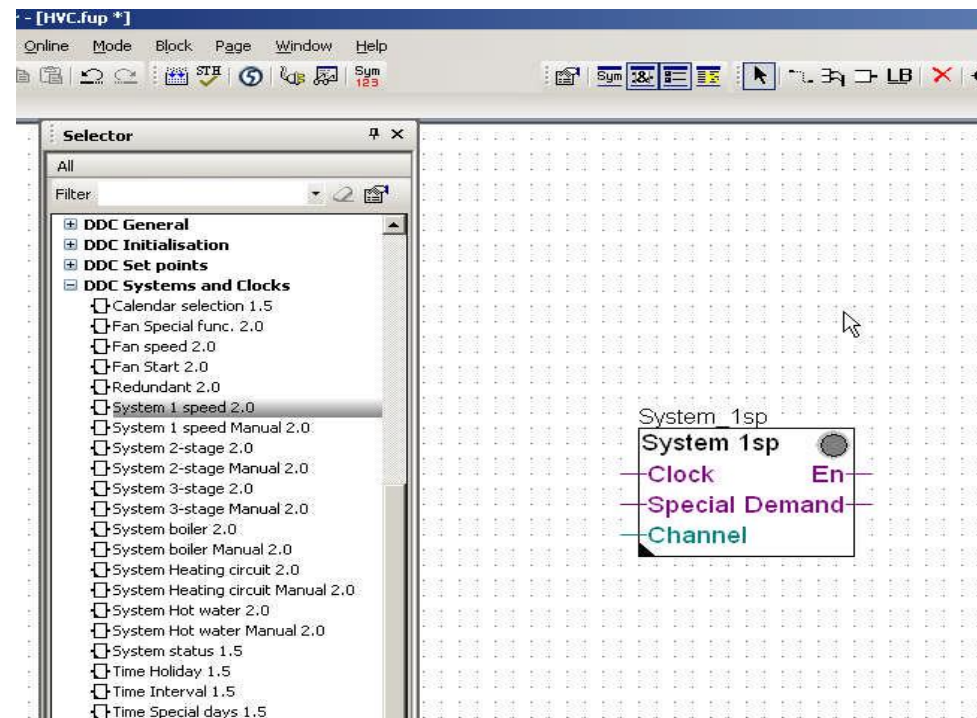Maybe you can also switch back automatically to the clock FBox if you detect that the SCADA is offline ...

Close adjust window. We do not need the symbol editor when writing the program by using FBoxes at the first moment. As you have seen the FBoxes will create automatically all resources by themselves.

With key "F5" it's easy to hide the symbol editor – pressing "F5" will pop up the symbol editor again.
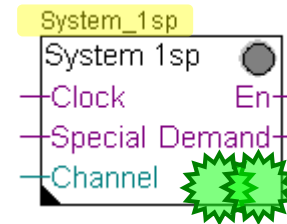
You should have this screen:

As you can see the FBox System 1sp has a default FBox name property System_1sp. DDC Suite FBoxes will use the FBox name property for some features therefore it's necessary to use a good name convention.

Double Click on FBox opens the Window: FBox properties.

Change the default name into S01_Switch

At the end you should have this:

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

In this example we use the clock feature via FBox input Clock. Therefore we have to add a clock FBox.

1. Select from FBox selector tab Application the family DDC Systems and Clocks.

2. Click on FBox Weekly Clock 1.5

3. Place FBox approx. at same position as you can see in picture (top/centered)

4. Connect the FBox output Channel with FBox input Clock.

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Now we have to close some FBox inputs. E.g. the input En from FBox WClock should be always high, the input Special from FBox System 1sp is not used, therefore always low.

1. Select from FBox selector tab Standard the family binary.

2. Use FBox High and connect at En from FBox WClock

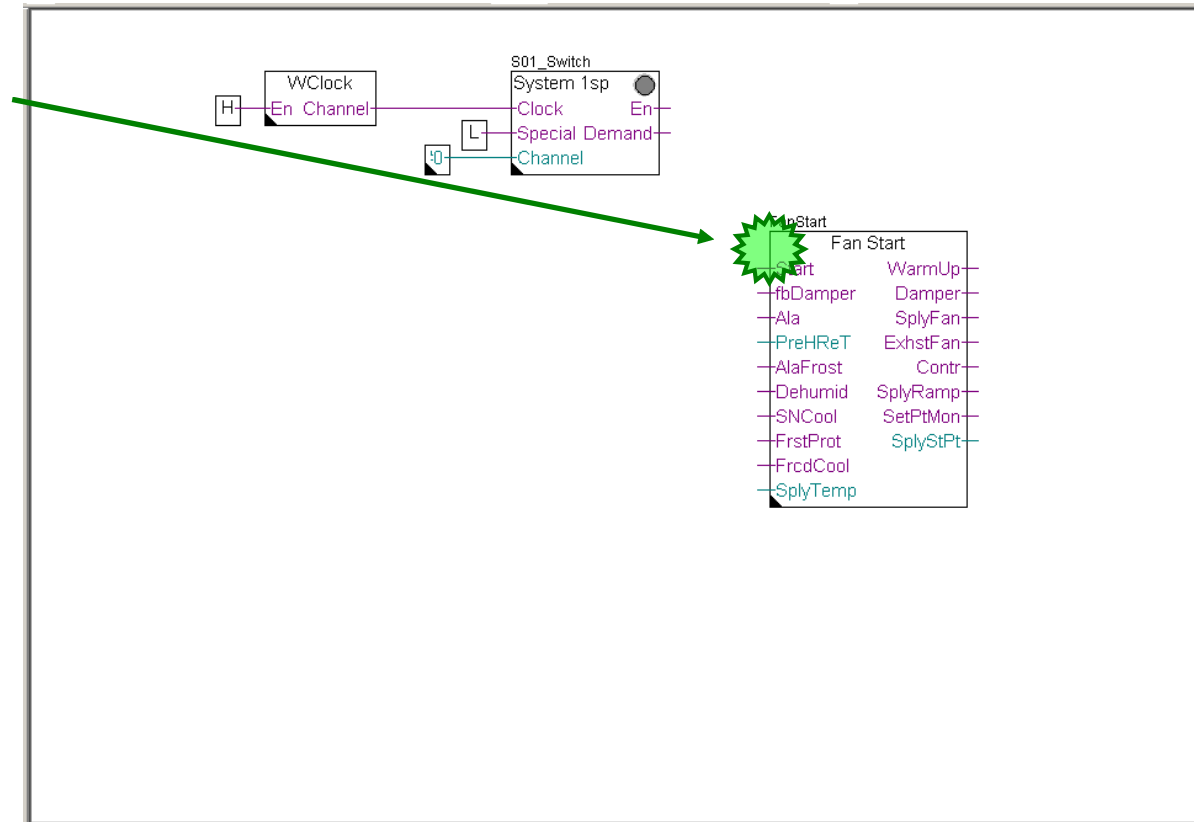3. Use FBox Low and connect at Special from FBox System 1sp

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

The input Channel from FBox System 1sp is also not used (optional for use with a calendar feature). Buth this is an integer connector, therefore we need a special FBox to "close" this connector.

1. Select from FBox selector tab Application the family DDC General.

2. Use FBox Register connect 1.5 and connect at Channel from FBox System 1sp

This FBox is also useful to set a constant value instead of a connector.

The input Channel from FBox System 1sp is also not used (optional for use with a calendar feature). Buth this is an integer connector, therefore we need a special FBox to "close" this connector.

1. Select from FBox selector tab Application the family DDC System and Clocks.

2. 2. Click on FBox Fan Start 2.0

3. Place FBox approx. at same position as you can see in picture

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Remember - the DDC Suite FBoxes will use the FBox name property for some features therefore it's necessary to use a good name convention.

Double Click on FBox opens the Window:

FBox properties:

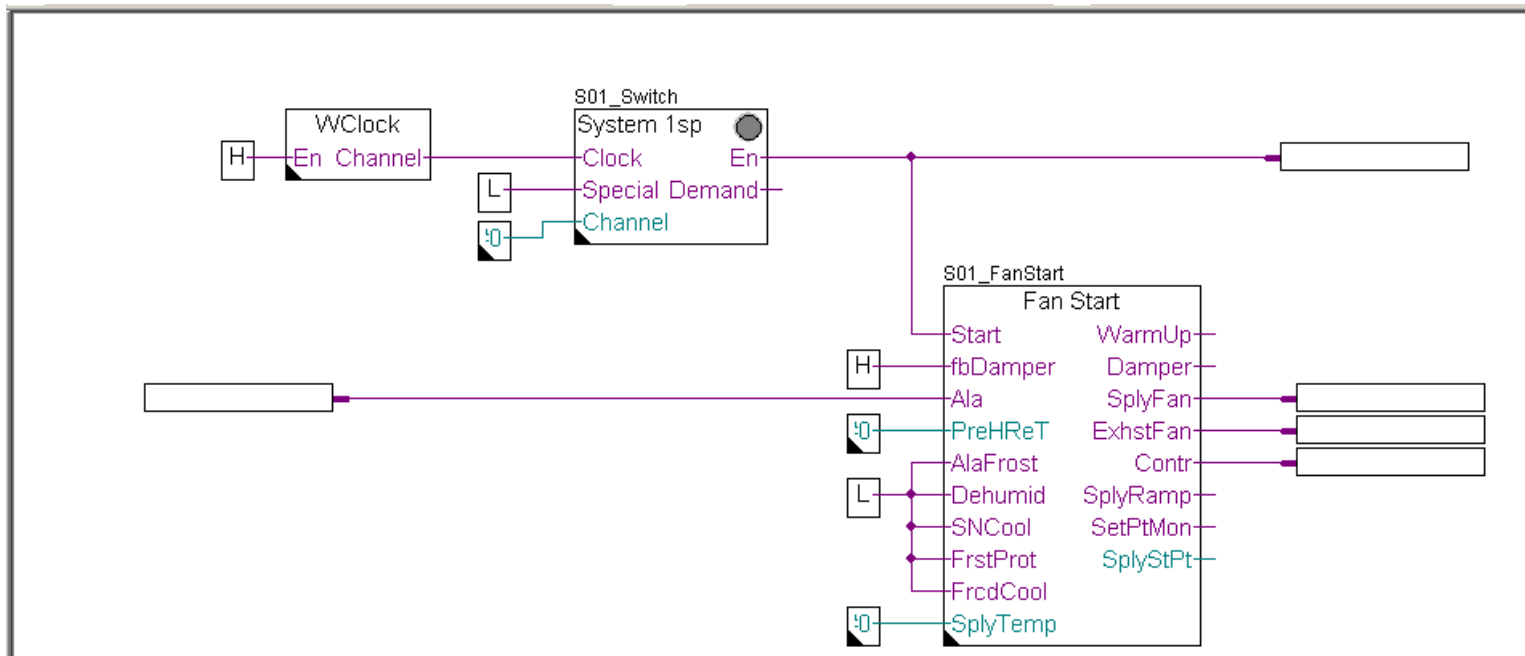Change the default name into S01_FanStart

At the end you should have this:

Now finish the this part by connecting some Low, High, Integer FBoxes and connectors.

FBox selector tab Standard, family Binary, FBox High and Low

FBox selector tab Application, family DDC General, FBox Register connect 1.5

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

The FBox Fan Start is a often used method to start up an air condition in a strongly defined sequence, e.g.

1. Warm up pre-heater to prevent a frost alarm when air condition will start in winter. If this has been successful proceed with

2. Open damper (flap) and wait until feedback "damper (flap) open"

3. Start supply air fan, wait some seconds and

4. Start exhaust air fan, wait some seconds and

5. Enable controlling (PID) an after some minutes

6. Start set point monitoring

There is also a set point ramp for supply air temperature to have a smooth start up of controller function to prevent a overreaction if the supply air is too warm after the warm up phase
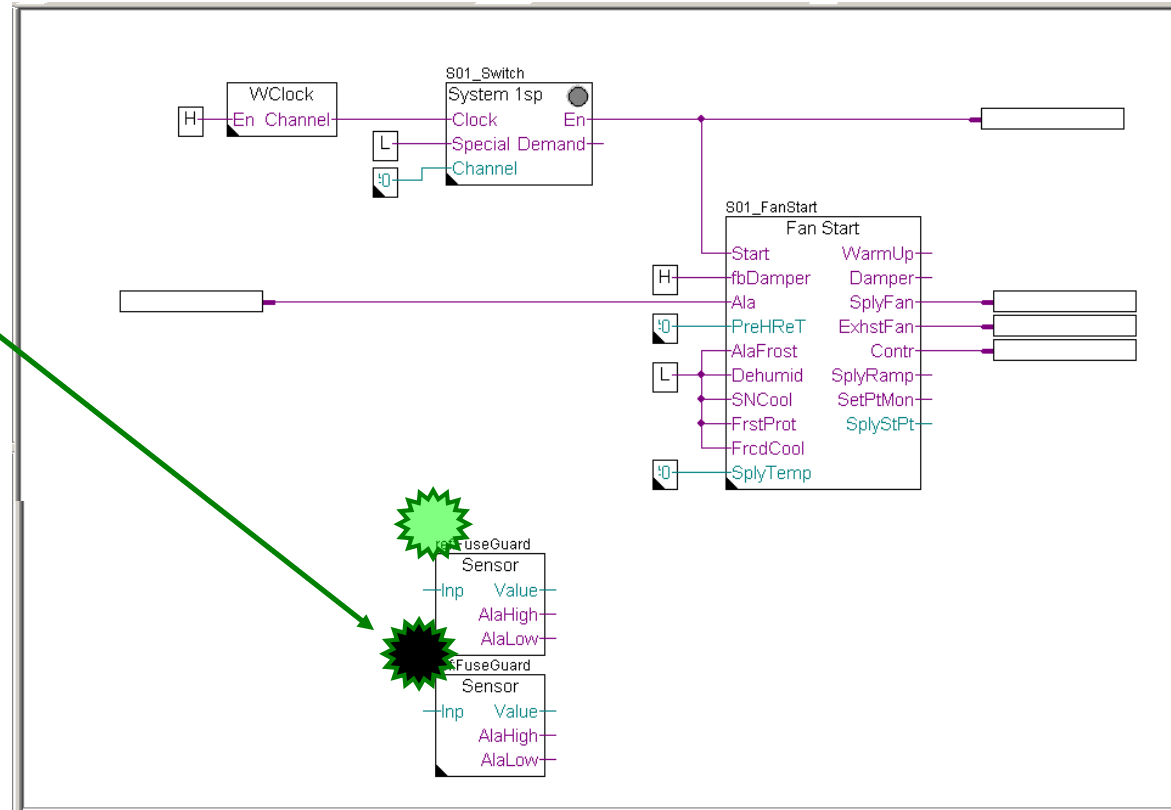
```
S01_FanStart
     Fan Start
─Start        WarmUp─
─fbDamper      Damper─
─Ala          SplyFan─
─PreHReT      ExhstFan─
─AlaFrost       Contr─
─Dehumid     SplyRamp─
─SNCool      SetPtMon─
─FrstProt     SplyStPt─
─FrcdCool
─SplyTemp
```

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Now we add 2 temperature sensors, the first should handle the supply air temperature, the second the exhaust air temperature

1. Select from FBox selector tab Application the family DDC System and Clocks.

2. 2. Click on FBox Fan Start 2.0

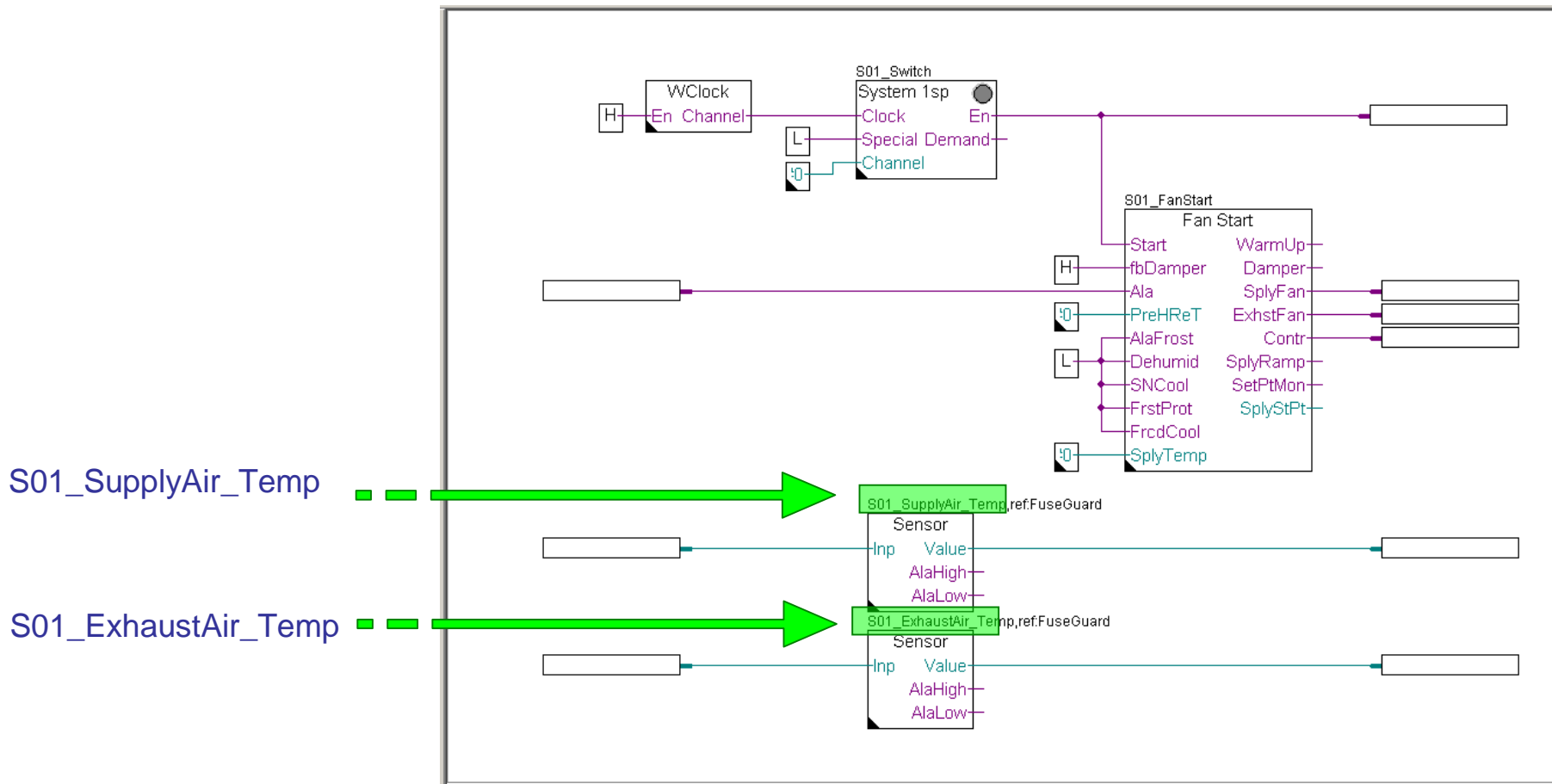3. Place 2 FBoxes approx. at same position as you can see in picture

Body text: "Now finish the this part by connecting the in and out connectors and editing the FBox name properties. The FBox reference properties stay unchanged."

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Now finish the this part by connecting the in and out connectors and editing the FBox name properties. The FBox reference properties stay unchanged.

Please enter the symbolnames as shown below

Please check in symbol editor (remember show/hide symbol editor with key "F5") if you see the same structure and symbols.

The Sensor FBox with name "S01_SupplyAirFan_Temp" is connected to symbols "iRawValue" and "iActualValue" because the input is the raw value from analogue input FBox and the output is the converted, filtered and calibrated actual value.

The Sensor FBox with name "S01_ExhaustAirFan_Temp" should be connected the same symbols but than we'll have e.g. the symbol "iRawValue" used for two different functionalities.

Now this is the point where we should start to structure the symbols we got from the FBoxes and declared by us.

Structuring data is good for
- Finding data in a big list much easier
- Gives more information about the data itself
- Reduces type writing error by reusing same symbol declarations
- Is the base for writing reusable software

- **structuring data**

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Let's edit the page property "name".
Right mouse click in the page in
Page Navigator select Properties
and type in text field "Name"

S01 Start/Stop air condition

Page navigator

- ⊞ ■ COB Allgemein
- ⊞ ■ COB Anlage_X
- ⊟ ■ COB System_Xa
  - 1: S01_Start/Stop AirCondition

Open Page	Enter
New Block	▶
Insert Page	Ins
Delete	Del
Cut	
Copy	
Paste	
Import Pages...	
Export Pages...	
Add to Templates...	
Add Side Connectors	
Remove Unused Connectors	
Move Up	Ctrl+Up
Move Down	Ctrl+Down
Expand All	
Collapse All	
Properties	

Properties

Page

General	
(Name)	S01_Start/Stop AirCondition
Comment	
Description	
Condition	
Disabled	No
Symbol	
Function	<
Value	0

Now we will start to organise the data in symbol editor - all from the FBoxes and the few symbols we defined are used in the air condition. Therefore they should be grouped in a "main" group named e.g. "S01" (=System 01).

S01

On this Fupla page we have 3 virtual functionalities, week clock, system switch and fan start. We can not touch them (not physically existing), therefore I recommend to define a sub group "System" within group "S01"

System

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Let's move the data from FBox WClock into the group S01.System. To find the data just click on FBox. Symbol editor will jump automatically into the group containing the first defined data from this FBox.

You see they are located in SystemClocks.WeekClock_0

Now drag&drop the group WeekClock_0 into the group S01.System

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Rename the group WeekClock_0 into

Weekclock

You can move and rename groups. This is useful to build up a clear structure and to define clear names. This will help you to find very easy data points in symbol editor – and the group names are used from Sweb and ViSi.Plus for mapping data from a FBox into a view object!

Creating a clear data structure is a must! The structure within this workshop is only a example how to do it.

But never move or rename the symbols within a group if they are created from a FBox automatically. They are a kind of data base name space convention. If you rename them the Sweb and ViSi.Plus view object won't work any more!

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Repeat the steps before also for FBox System 1sp. Click on FBox, drag&drop group System1_0 into group S01.System.

Rename group System1_0 into

Switch

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Repeat the steps before also for FBox Fan Start. Click on FBox, drag&drop group FanStart_0 into group S01.System.

Rename group FanStart_0 into

FanStart

Now the symbols in connectors connected to the FBox inputs or outputs at this 3 FBoxes should also be moved into the group S01.System.

Mark the symbols:

- iStartSystem

- iStartSupplyAirFan

- iStartExhaustAirFan

- iStartController

- iGroupAlarm

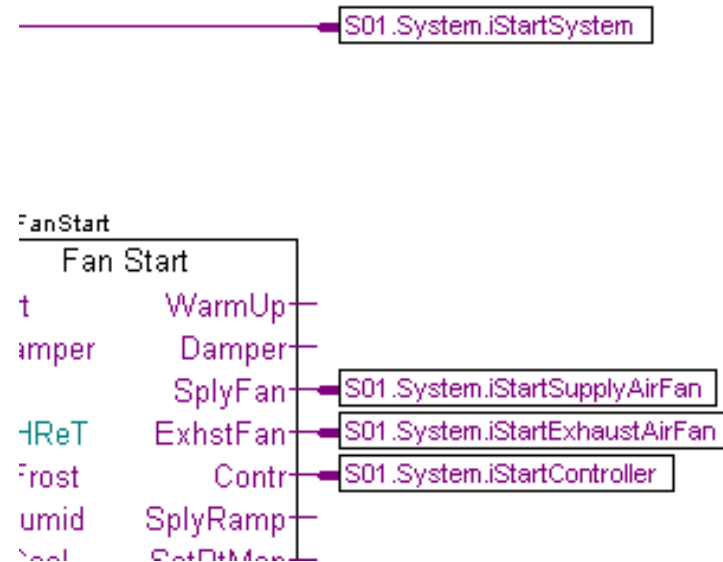
And drag&drop them into group S01.System

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

When renaming in symbol editor the symbols in the input or output connectors will be automatically updated.

So you have also a better identification in your program when reading the symbols.

At this page there are 2 Sensor FBoxes left. One supports the supply air temperature sensor and the other the exhaust air temperature sensor.

Maybe there will be more available in the supply air, e.g. the supply air fan or a supply air humidity sensor – therefore I recommend to create a subgroup SupplyAir within group S01 and also a subgroup Temperature within subgroup SupplyAir.

Same for Exhaust Air …

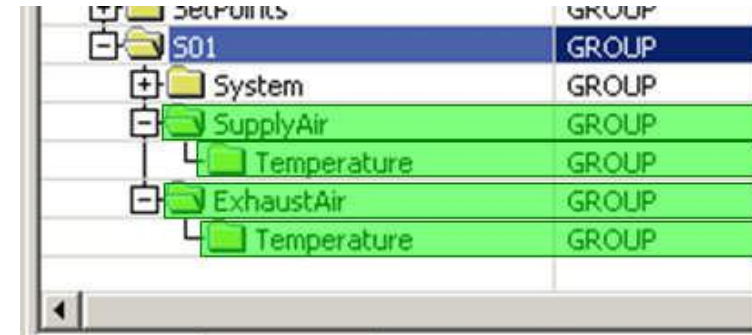I recommend a depth of 4 groups:
1st group = the system, e.g. AC01 (=Air Condition 01)
2nd group = where it's located, e.g. SupplyAir
3rd group = what's there, e.g. Temperature
4th group = the functionality or component, e.g. Sensor

When reading the group structure AC01.ExhaustAir.Humidity.Sensor you will know the location at the plant and will find very fast all infiormation depending to this sensor in the symbol editor.

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Repeat the steps before also for FBox Sensor with name S01_SupplyAir_Temp. Click on FBox, drag&drop group Sensor_0 into group S01.SupplyAir.Temperature.

Rename the group Sensor_0 into Sensor.

Repeat the steps before also for FBox Sensor with name S01_ExhaustAir_Temp. Click on FBox, drag&drop group Sensor_1 into group S01.ExhaustAir.Temperature.

Rename the group Sensor_1 into Sensor.

Now the symbols in connectors connected to the FBox inputs or outputs at the first FBox should also be moved into the group S01.SupplyAir.Temperatur.

Mark the symbols:

- iRawValue

- iActualVal

And drag&drop them into group S01.SupplyAir.Temperatur

The second FBox also need the same symbols in the group of S01.ExhaustAir.Temperatur.

But we did not declared to prevent having same symbols used for different functions.

Instead of creating them now manually we duplicate them in SymbolEditor.

- mark both symbols in group S01.SupplyAir.Temperatur.
- press "Ctrl" key and drag&drop them into group S01.ExhaustAir.Temperatur

With pressing "Ctrl" key we duplicate the symbols! Without you move them from one group into another.

Be always aware if you want to move or duplicate symbols!

S01	GROUP
System	GROUP
SupplyAir	GROUP
Temperature	GROUP
iRawValue	R
iActualVal	R
Sensor	GROUP
ExhaustAir	GROUP
Temperature	GROUP
Sensor	GROUP

S01	GROUP
System	GROUP
SupplyAir	GROUP
Temperature	GROUP
iRawValue	R
iActualVal	R
Sensor	GROUP
ExhaustAir	GROUP
Temperature	GROUP
iRawValue	R
iActualVal	R
Sensor	GROUP

Now just drag&drop the new symbols into the connector in Fupla page.

Doing it this way you'll have always the same name convention for same functionalities.
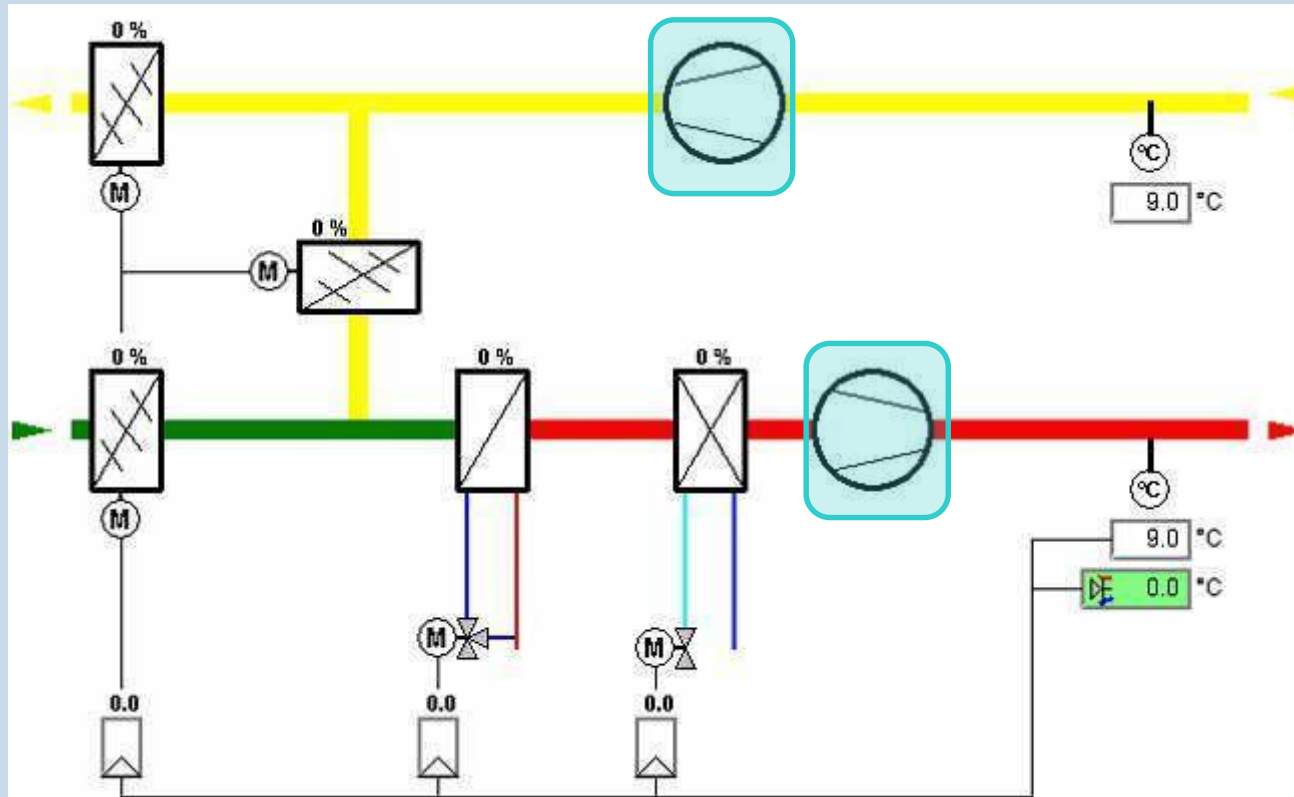
DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Second Fupla page will contain

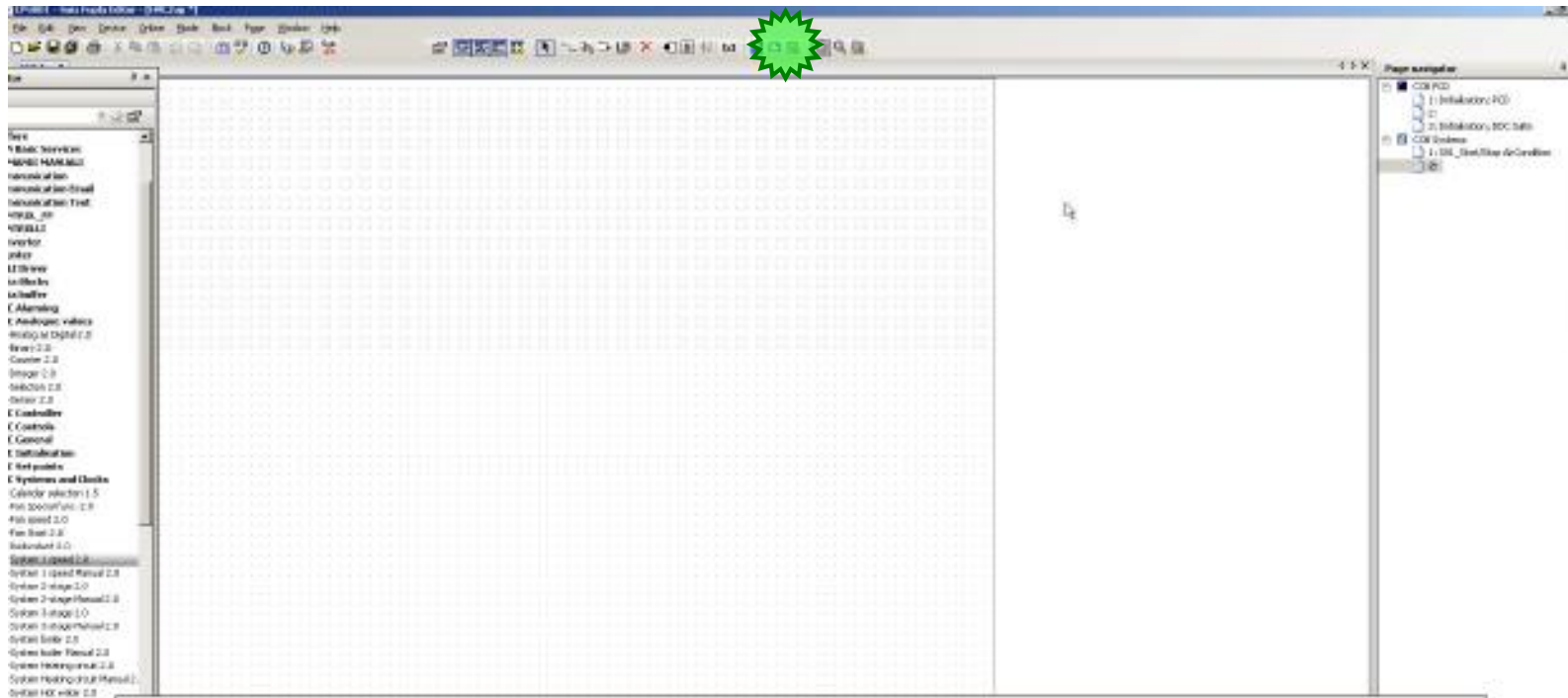- physical components supply air fan, exhaust air fan
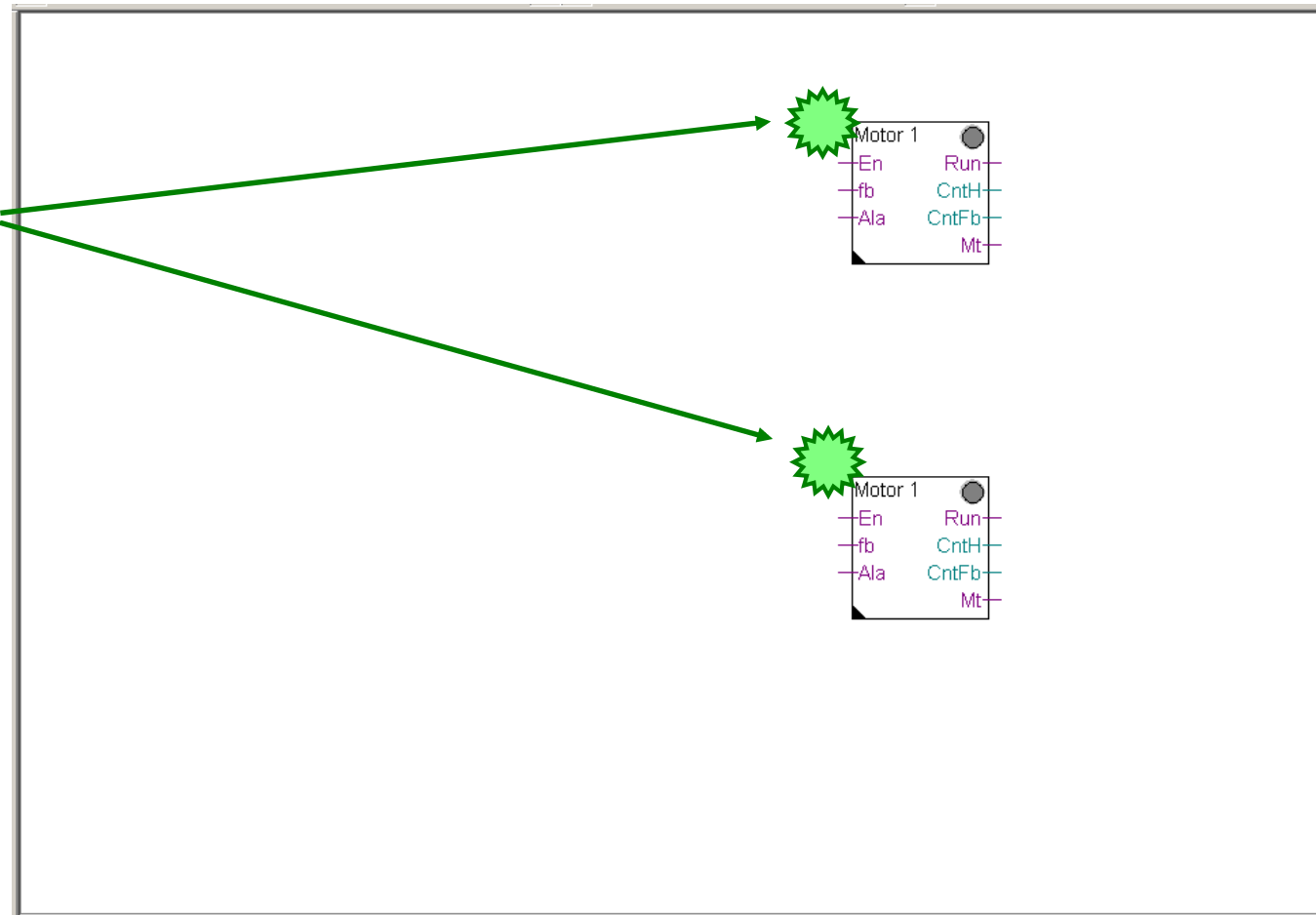
Add a new page after the current page

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

First we use some control FBoxes

1. Select from FBox selector tab Application the family DDC Controls

2. 2. Click on FBox Motor 1 speed 2.0

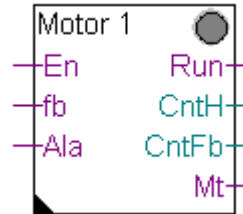3. Place 2 FBoxes approx. at same position as you can see in picture

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

The FBox Motor 1 speed controls any drive via one digital output. The features are

- Virtual switch e.g. to start/stop for testing or maintenance
- Start delay
- Counting working hours
- Counting switch-on via feedback input
- Monitoring this counter to indicate e.g. after 2000 working hours "maintenance necessary"
- Collecting all information for a clear presentation why the motor is running or not

Motor 1

En	Run
fb	CntH
Ala	CntFb
	Mt

Properties

DDC Controls:Motor 1 speed 2.0

General	
(Name)	
Comment	

Adjust Parameters

System functions

| PCD Alarm administration (Inde... | 0 |
| BACnet | No |

Settings

Digital output	-1
HMI Lower prio...	Automatic
Start delay (s)	0.0

Counting

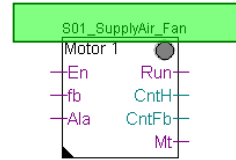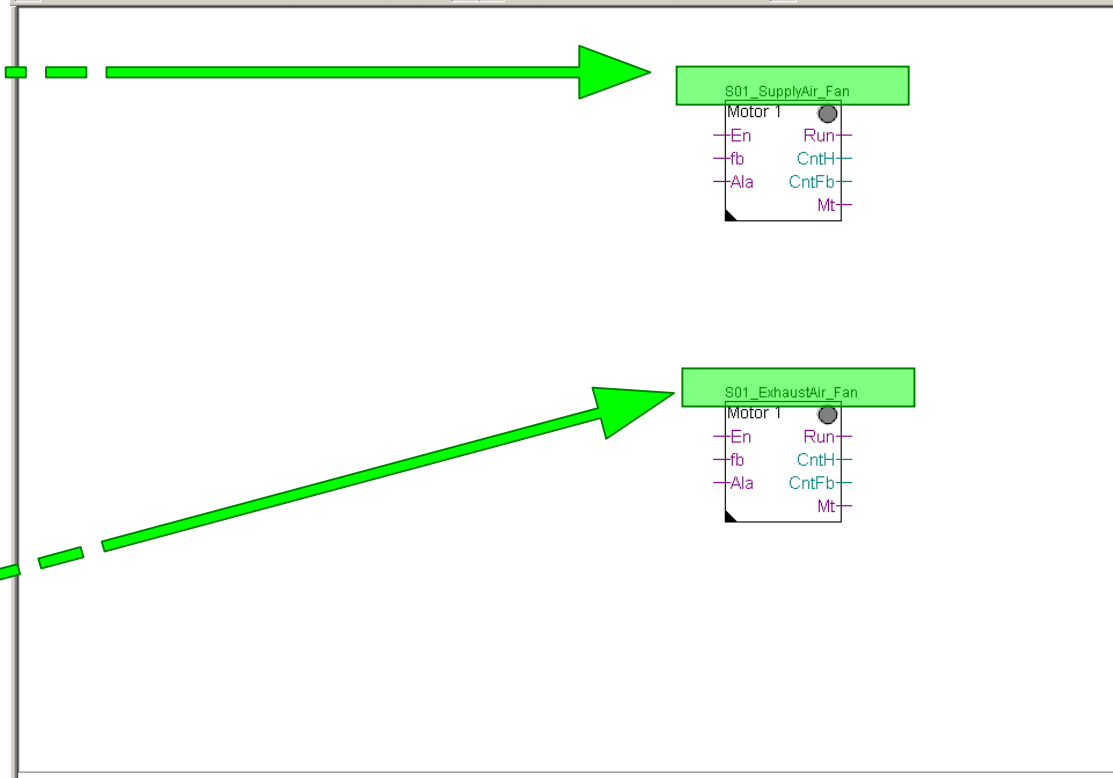| Feedback | 0 |
Message after feedback	2000
Hours	0
Message after hours	5000

Again: the FBox name property is used for some features. Give each FBox a clear name.

(FBox at top)

S01_SupplyAir_Fan

(FBox at bottom)

S01_ExhaustAir_Fan

Now we the alarm monitoring

1. Select from FBox selector tab Application the family DDC Alarming

2. Click on FBox Motor 1 speed 2.0

3. Place 2 FBoxes approx. at same position as you can see in picture
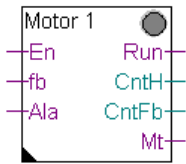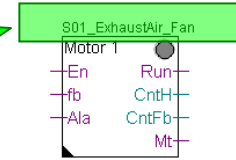
Give each FBox a clear name.

(FBox at top)

S01_SupplyAir_FanAla

(FBox at bottom)

S01_ExhaustAir_FanAla

DDC Suite 2.0 / PG5 Building Advanced
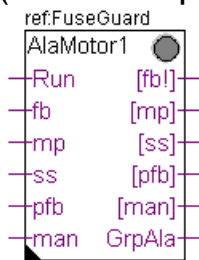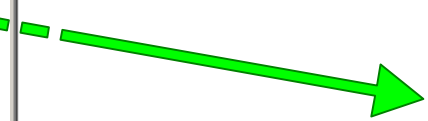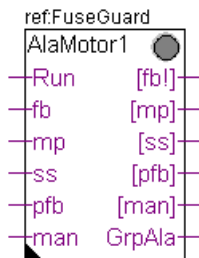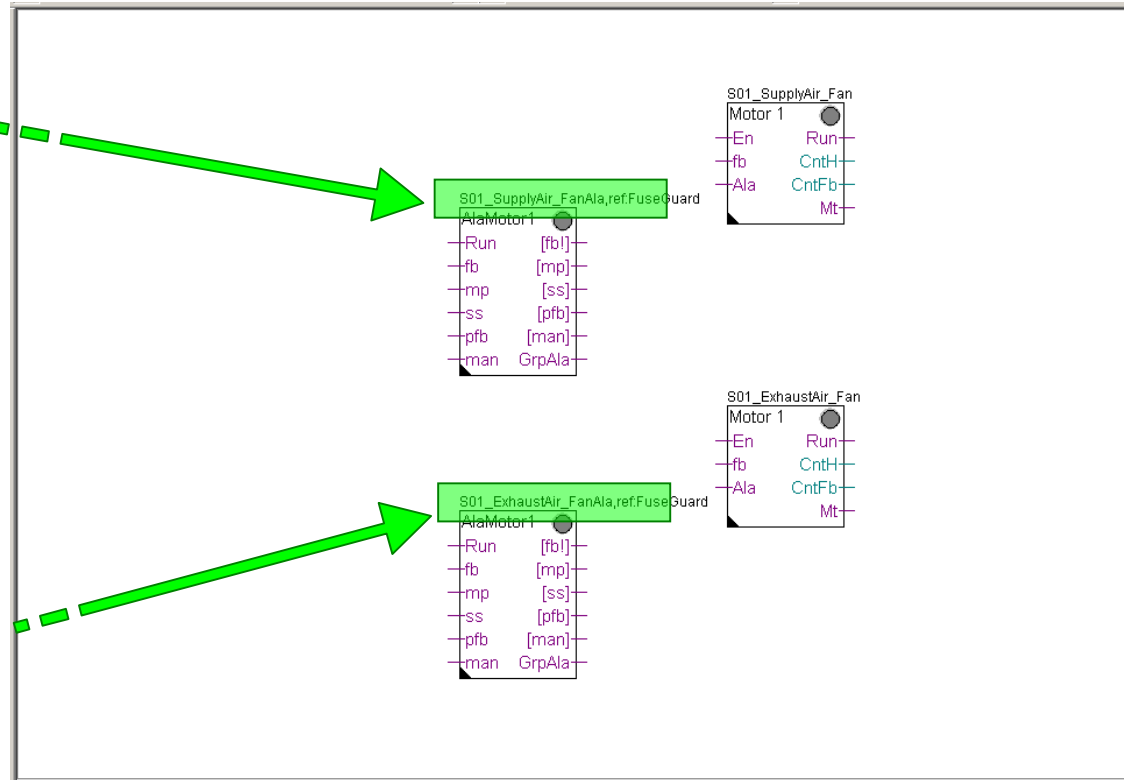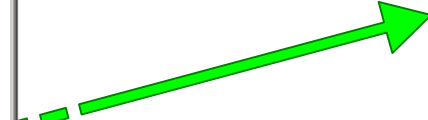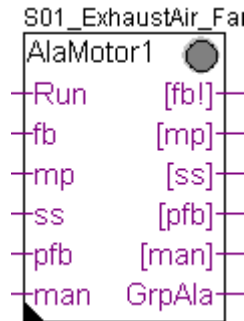## Working with Fupla

The FBox AlaMotor1 is monitoring at least 5 typically alarms a motor can have. The alarms could be

- Feedback (missing)
- Process feedback (missing)
- Motor protection
- Service switch
- Manual override switch

We have to modify the parameter "Normal input state" in group "[--- Process feedback ---]". By default it's "opened" – modify this into

Closed

Do this also in the second AlaMotor1 FBox.

S01_ExhaustAir_Fan

AlaMotor1
- Run   [fb!]
- fb   [mp]
- mp   [ss]
- ss   [pfb]
- pfb   [man]
- man   GrpAla

Properties

DDC Alarming:Motor 1 speed 2.0

General	
(Name)	
Reference	FuseGuard
Comment	

Adjust Parameters	
System functions	
PCD Alarm administration (Inde...	0
BACnet	No
Group alarm from fb/mp/pfb	Only these
Feedback	
Digital input	-1
Delay	5.0
Process feedback	
Digital input	-1
Normal input state	Closed
Delay (Sec)	30.0
Motor protection	
Digital input	-1
Acknowledgement mandatory	No
Normal input state	opened
Alarm suppression	for appl. vltg.
Service switch	
Digital input	-1
Acknowledgement mandatory	No
Normal input state	opened
Alarm suppression	for appl. vltg.
Manual override	
Digital input	-1
Normal input state	opened
Alarm suppression	for appl. vltg.

Now finish the this part by connecting some Low, Or FBoxes and connectors.

Finally we add the FBox selector tab Application, family DDC Alarming FBox Acknowledge 1.5.

Via this FBox we can acknowledge stored alarms from this position till end of program. It's easier than switching to initialization page to access the adjust window from InitLib FBox.

Show symbol editor (key "F5") and drag&drop some symbols from symbol editor into connectors

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Please check in symbol editor (remember show/hide symbol editor with key "F5") if you see the same structure and symbols.

- **structuring data**

Double click in Fupla page and type in

text field "Name"

S01 Supply/Exhaust air fan

At this page there are 2 AlaMotor1 FBoxes and 2 Motor 1.
One pair supports the supply air fan and the other the
exhaust air fan.

In S01 there is already a subgroup SupplyAir and
ExhaustAir – we have just to add there a subgrup Fan.
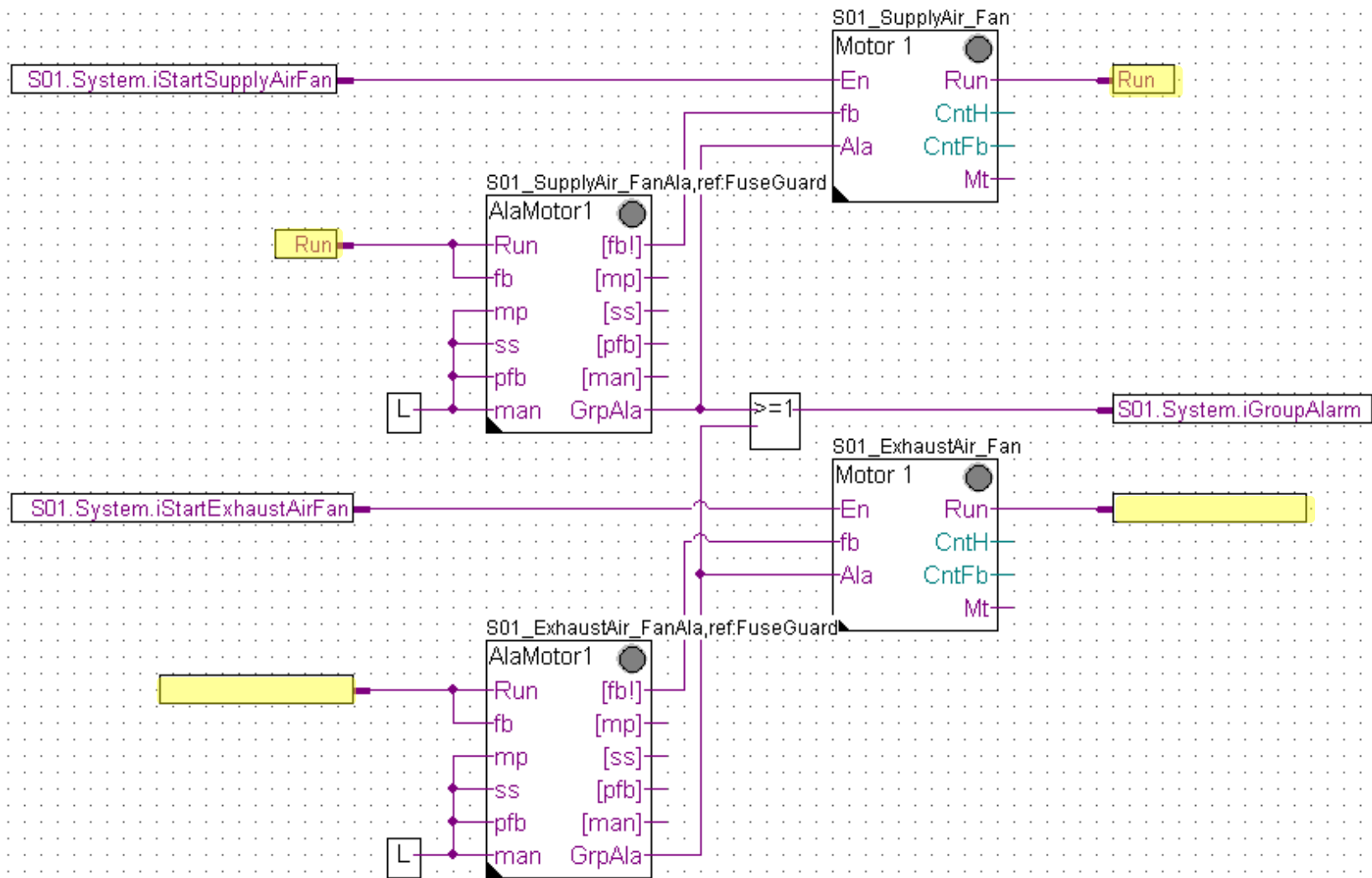

Reminder: I recommend a depth of 4 groups:
1st group = the system, e.g. AC01 (=Air Condition 01)
2nd group = where it's located, e.g. SupplyAir
3rd group = what's there, e.g. Temperature
4th group = the functionality or component, e.g. Sensor

When reading the group structure AC01.ExhaustAir.Humidity.Sensor you will know the location at the
plant and will find very fast all infiormation depending to this sensor in the symbol editor.

Repeat the steps before also for FBox Motor 1 with name S01_SupplyAir_Fan. Click on FBox, drag&drop group Motor1speed_0 into group S01.SupplyAir.Fan.

Rename the group Motor1speed_0 into Control. (because the fan will have a second FBox and this FBox will control the fan)

Repeat the steps before also for FBox AlaMotor1 with name S01_SupplyAir_FanAla. Click on FBox, drag&drop group Motor1_0 into group S01.SupplyAir.Fan.

Rename the group Motor1_0 into Alarming. (because the fan has already another FBox and this FBox will handle the alarming of the fan)

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Repeat the steps before also for the exhaust air fan.

Symbols

Group/Symbol	Type
S01	GROUP
System	GROUP
SupplyAir	GROUP
ExhaustAir	GROUP
Temperature	GROUP
Fan	GROUP
Control	GROUP
Alarming	GROUP

The symbol in connectors for Supply Air Fan
should also be moved into the group
S01.SupplyAir.Fan.

Mark the symbol iSupplyAirFanRun and
drag&drop them into group S01.SupplyAir.Fan

Group/Symbol	Type
Run	F
PCD	GROUP
S01	GROUP
System	GROUP
SupplyAir	GROUP

Group/Symbol	Type
PCD	GROUP
S01	GROUP
System	GROUP
SupplyAir	GROUP
Temperature	GROUP
Fan	GROUP
Run	F
Control	GROUP
Alarming	GROUP

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

The second FBox also need the same symbol in the group of S01.ExhaustAir.Fan.

But we did not declared to prevent having same symbols used for different functions.

Instead of creating them now manually we duplicate them in SymbolEditor.

- mark symbol in group S01.SupplyAir.Fan.
- press "Ctrl" key and drag&drop them into group S01.ExhaustAir.Fan

With pressing "Ctrl" key we duplicate the symbols! Without you move them from one group into another.

Be always aware if you want to move or duplicate symbols!

Group/Symbol	Type
⊟📁	
⊞📁 PCD	GROUP
⊟📁 S01	GROUP
⊞📁 System	GROUP
⊟📁 SupplyAir	GROUP
⊞📁 Temperature	GROUP
⊟📁 Fan	GROUP
Run	F
⊞📁 Control	GROUP
⊞📁 Alarming	GROUP

Group/Symbol	Type
⊟📁	
⊞📁 PCD	GROUP
⊟📁 S01	GROUP
⊞📁 System	GROUP
⊟📁 SupplyAir	GROUP
⊞📁 Temperature	GROUP
⊟📁 Fan	GROUP
Run	F
⊞📁 Control	GROUP
⊞📁 Alarming	GROUP
⊟📁 ExhaustAir	GROUP
⊞📁 Temperature	GROUP
⊟📁 Fan	GROUP
Run	F
⊞📁 Control	GROUP
⊞📁 Alarming	GROUP

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Now just drag&drop the new symbol into the connector in Fupla page.

Doing it this way you'll have always the same name convention for same functionalities.

Title of this presentation

72

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Third Fupla page will contain

- physical components controller cooler, controller mixed air, controller pre-heater

Add a new page after the current page

First we use some control FBoxes

1. Select from FBox selector tab Application the family DDC Controller

2. Place FBox Cooler 2.0

3. Place FBox Mixed air 2.0

4. Place FBox Preheater 2.0

Give each FBox a clear name. FBox reference properties get a prefix S01_.

Name: S01_Cooler
Reference: S01_FanStart

Name: S01_MixedAir
Reference: S01_FanStart

Name: S01_Preheater
Reference: S01_FanStart

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Now finish the this part by connecting some Low, Register connectors and Value FBoxes.

1. Select from FBox selector tab Application the family DDC Set points

2. Place FBox Integer 2.0

3. Type in FBox name property S01_SupplyAirTempSetPoint

[L] FBox selector tab Standard, family Binary, FBox Low

[:0] FBox selector tab Application, family DDC General, FBox Register connect 1.5

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

We have to set a value in FBox Register connect connected to Input YCldStrt:

1. Open adjust window with double click on FBox

2. Type in value 456

This represents 45.6 and will be used as valve signal during start up phase. More info later on during test ….

S01_Preheater,ref:S01_FanStart
ref:FanStart
PreHeater
EnContr   Y
SetPt   Act
Temp   SetPt
YCldStrt
-      ????
PH   ????
+   ????

Properties   ⊓ ×

DDC General:Register connect 1.5

⊟ **General**

| (Name) | |
| Comment | |

⊟ **Adjust Parameters**

Value at output #0	456
Value at output #1	0
Value at output #2	0
Value at output #3	0
Value at output #4	0

⊟ **Advanced Info**

| Name | Register connect 1.5 |
| Macro Name | _DDC_GECONNECT01 |

Value at output #0

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Now finish the this part by connecting input and output connectors.

Show symbol editor (key "F5") and drag&drop some symbols from symbol editor into connectors

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

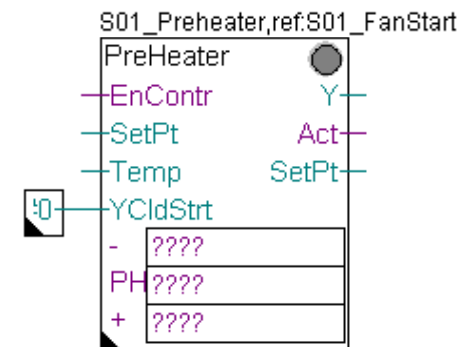# DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Please check in symbol editor (remember show/hide symbol editor with key "F5") if you see the same structure and symbols.

There are still some connectors undefined, displaying "????". Within DDC Suite the controller sequence is not defined within 1 FBox (e.g. like the HMC FBox from Heavac library).

The sequence is defined by creating a "chain" (handshaking) between the controller FBoxes. Therefore each FBox provides 2 or 3 connectors to define some flags for "chaining".

- The connector with a shortcut for FBox name, e.g. C = Cooler, MA = MixedAir or PH = PreHeater identifies a Flag which is monitoren by the FBox itself. If this Flag is high (and also the input EnContr) the controller will work.

- The connector with a "-" (minus) identifies the flag if a controller which should be activated when this controller FBox calculates less than 2 % signal (let's say "need less energy in air")

- The connector with a "+" (minus) identifies the flag if a controller which should be activated when this controller FBox calculates more than 98 % signal (let's say "need more energy in air")

Lets start with the FBox Cooler:

- C connector = iCooler. This flag is monitored by FBox

FBox MixedAir:

- MA connector = iMixedAir. This flag is monitored by FBox.

FBox PreHeater:

- PH connector = iPreHeater. This flag is monitored by FBox.

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Lets start with the FBox Cooler:
- **C connector** is already defined (own Flag)
- **"+" connector** = iMixedAir – because if signal from Cooler is less than 2 % the MixedAir should be activated

FBox MixedAir:
- **MA connector** is already defined (own Flag)
- **"-" connector** = iCooler – because if signal from MixedAir is less than 2 % the Cooler should be activated again
- **"+" connector** = iPreHeater – because if signal from MixedAir is bigger than 98 % the PreHeater should be activated.

FBox PreHeater:
- **PH connector** is already defined (own Flag)
- **"-" connector** = iMixed – because if signal from PreHeater is less than 2 % the MixedAir should be activated again
- **"+" connector** = iReHeater – because if signal from PreHeater is bigger than 98 % the ReHeater should be activated. (OK – in this example we don't have – but that's no problem!)

S01_Cooler,ref:S01_FanStart
Cooler
EnContr Y
SetPt Act
Temp SetPt
Lock
C iCooler
+ iMixedAir

S01_MixedAir,ref:S01_FanStart
MixedAir
EnContr Y
SetPt YInv
Temp Act
RoomTemp SetPt
- iCooler
MA iMixedAir
+ iPreHeater

S01_Preheater,ref:S01_FanStart
PreHeater
EnContr Y
SetPt Act
Temp SetPt
YCldStrt
- iMixedAir
PH iPreHeater
+ IReHeater

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

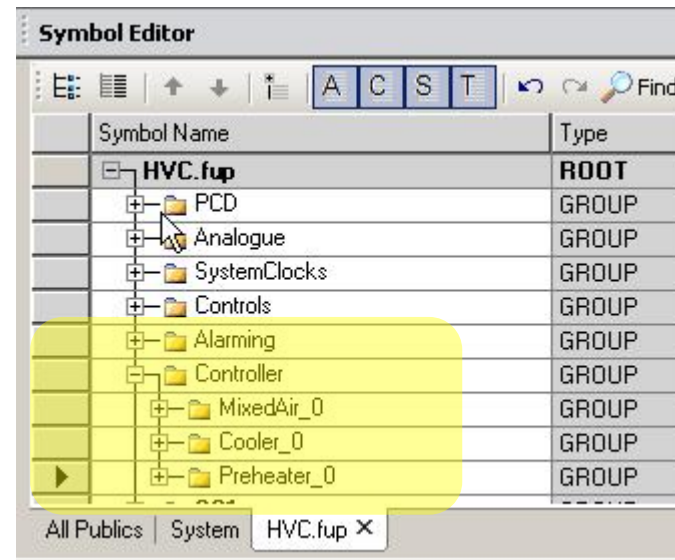Please check in symbol editor (remember show/hide symbol editor with key "F5") if you see the same structure and symbols.
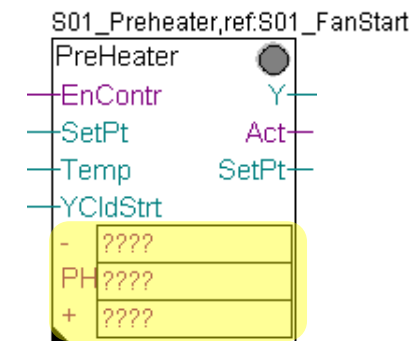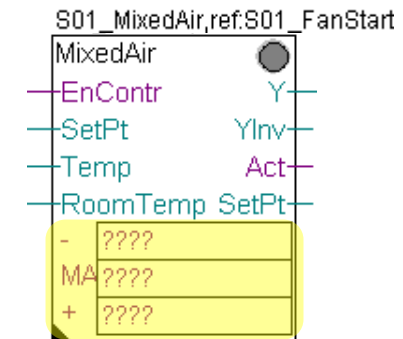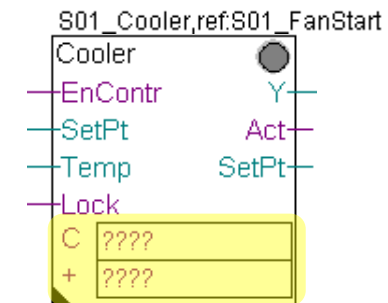
DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

At the end we have a small air condition but a look into the symbol editor will display a lot of symbols - 98% of them are created automatically by placing a FBox into Fupla page.

The missing 2% are user defined and only necessary to connect information between Fupla pages ore FBoxes.

Let's organise this big amount of data into a clear structure. The target is to get a data structure where it's easy to find any information and also to get a reusable air condition template.

- **structuring data**

Double click in Fupla page and type in

text field "Name"

S01 Controller

At this page there are 3 controller FBoxes representing an unit and 1 FBox for a set point. Maybe the unit will have more than one FBox it's always a good thing to create a separate group

So let's create a group

Cooler to store all data for the cooler/valve unit

MixedAir to store all data for the mixed air/damper unit

Preheater to store all data for the preheater/valve unit

Repeat the steps before also for FBox Cooler with name S01_Cooler. Click on FBox, drag&drop group Cooler_0 into group S01.Cooler. Rename the group Cooler_0 into Controller.

Repeat the steps before also for FBox MixedAir with name S01_MixedAir. Click on FBox, drag&drop group MixedAir_0 into group S01.MixedAir. Rename the group MixedAir_0 into Controller.

Repeat the steps before also for FBox PreHeater with name S01_Preheater. Click on FBox, drag&drop group Preheater_0 into group S01.Preheater. Rename the group Preheater_0 into Controller.

1. Mark the symbol iCooler drag&drop it into group S01.Cooler

2. Mark the symbol iMixedAir drag&drop it into group S01.MixedAir

3. Mark the symbol iPreHeater and iReHeater drag&drop them into group S01.Preheater

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

The symbol in connectors for Cooler should also be moved into the group S01.Cooler.

Mark the symbol iSignal and drag&drop it into group S01.Cooler

The other controller FBoxes also need the same symbol in the group of S01.MixedAir and S01.PreHeater.

But we did not declared to prevent having same symbols used for different functions.

Instead of creating them now manually we duplicate them in SymbolEditor.

- mark symbol in group S01.Cooler.
- press "Ctrl" key and drag&drop them into group S01.MixedAir

Repeat this also for PreHeater.

With pressing "Ctrl" key we duplicate the symbols! Without you move them from one group into another.

Be always aware if you want to move or duplicate symbols!

Group/Symbol	Type
⊟ 🗁	
⊞ 🗀 PCD	GROUP
⊟ 🗀 S01	GROUP
⊞ 🗀 System	GROUP
⊞ 🗀 SupplyAir	GROUP
⊞ 🗀 ExhaustAir	GROUP
⊟ 🗀 Cooler	GROUP
─ 🗀 iCooler	F
─ 🗀 iSignal	R
⊞ 🗀 Controller	GROUP

Group/Symbol	Type
⊟ 🗁	
⊞ 🗀 PCD	GROUP
⊟ 🗀 S01	GROUP
⊞ 🗀 System	GROUP
⊞ 🗀 SupplyAir	GROUP
⊞ 🗀 ExhaustAir	GROUP
⊟ 🗀 Cooler	GROUP
─ 🗀 Cooler	F
─ 🗀 iSignal	R
⊞ 🗀 Controller	GROUP
⊟ 🗀 MixedAir	GROUP
─ 🗀 iMixedAir	F
─ 🗀 iSignal	R
⊞ 🗀 Controller	GROUP
⊟ 🗀 PreHeater	GROUP
─ 🗀 iPreheater	F
─ 🗀 iReHeater	F
─ 🗀 iSignal	R
⊞ 🗀 Controller	GROUP

Now we just have to drag&drop them from SymbolEditor into Fupla connector

At least we have to move FBox Value with name S01_SupplyAirTempSetPoint. Click on FBox, drag&drop group Integer_0 into group S01.SupplyAir.Temperature. Rename the group Integer_0 into SetPoint

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

At least all data should be moved into group S01(or subgroups) and no symbol any more in root

The groups Analogue, Controls, Alarming and Controller should also be empty (no "+" sign in front of the folder)

Now we have moved all data in a clear and unique structure.

Now press key "F2" to build the program.

Any error messages?

Yes: rewind to first slide and repeat all lessons …

At this point we have created a small and nice air condition application. In real life it would be bigger with much more FBoxes or symbols – but we can reuse this in future if we store it as template.

Therefore we export this application as tempalte.
Click on Page in menu and in context menu entry
Export…

Type in file name, use AirCondition and press
button Export

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Within Dialog Export Pages select option All and finish with button OK.

End of engineering a Fupla application. With DDC Suite FBoxes we reduce the manual work to define symbols for FBoxes – only the symbols in the side connectors must be defined manually.

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

4th Fupla page (and last page …) will contain

- physical layer for testing procedure

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

Add a new page after the current page

DDC Suite 2.0 / PG5 Building Advanced
Working with Fupla

1. Select from FBox selector
   tab Application the family
   HVC Analogue

2. Place FBox PCD2.W4
   stretch up to 2 inputs

3. Place FBox PCD2.W2
   stretch up to 2 outputs

Connect also to FBox inputs and outputs all connectors

1. Type O 48 into PCD2.W4 FBox connector

2. Type I 32 into PCD2.W2 FBox connector

3. drag&drop some symbols from symbol editor into connectors

Now press key "F2" to build the program.

Any error messages?

Yes: rewind to first slide and repeat all lessons …

Everything OK: download program

DDC Suite - Online features

After download and run we will demonstrate the online features of DDC Suite.

All parameter in DDC Suite FBoxes are online parameter, that means that also some basic settings can be made online without forcing

- Go offline
- change parameter
- compile
- download

This will reduce commissioning time

If PCD clock is on time the FBox System 1sp will enable the air condition by clock demand.

Until all settings have been made we'll block the air condition, just open adjust window and set parameter "HMI Lower priority" to Off and write this into PCD.

You will see that the LED of the FBox turns to RED – indication that this "system" is under manual operation.

Close adjust window.

DDC Suite 2.0 / PG5 Building Advanced
Online features

Lets have a look into FBox Sensor – open adjust window from FBox with name S01_SupplyAir_Temp.

By default this FBox expects already a physical value at input In. But the PCD workshop model supports only a active linear signal – so we have to convert the value. Therefore

- Select Conversion at parameter CardType
- set Physical. Value min to 15.0
- set Physical. Value min to 26.0
- set raw input value min to 0
- set raw input value max to 1000

And write this settings into PCD.

Also the exhaust air temperature sensor must be parameterized. Open adjust window from FBox with name S01_ExhaustAir_Temp.

This sensor may have another linearization. Therefore

- Select Conversion at parameter CardType
- set Physical. Value min to 10.0
- set Physical. Value min to 40.0
- set raw input value min to 0
- set raw input value max to 1000

And write this settings into PCD.

You see you can adapt very easy the linearization for a sensor e.g. if a sensor must be replaced and the physical or/and raw values are different.

Additional a correction can be made, also a filter and last but not least a limit low/high is monitored. E.g. if you have a supply air temperature sensor you can define 5.0 as low and 70.0 as high limit to indicate a "short wire" or "broken wire".

On page S01 Supply/Exhaust air fan we can parameterize the fan. Open adjust window from FBox with name S01_ExhaustAir_Fan

You see parameter Digital output contains -1 – this means no hardware output controlled from this FBox. We can access any digital output simply by typing in the output address – type in 16 and write into PCD.

Now we can manually start/stop the fan by setting parameter HMI Lower priority to On or Off. The FBox will set FBox output Run to required state – but also the defined hardware output.

All FBoxes normally controlling a digital output are able to access the hardware output by themselves. If no output should be accessed use -1 – then no hardware address is defined and this option is disabled.

Please set parameter HMI Lower priority back to Automatic and write into PCD.

Open adjust window from FBox with name S01_ExhaustAir_FanAla

Here we can define the input handling for typically motor alarms. If the parameter Digital input contains -1 the FBox input is used, as soon the value is set to a real address this input is used.

Additional a mandatory acknowledge can be set, that means if the input turns to alarm state an back to normal state the alarm is still active and must be acknowledged (use e.g. the FBox Ack in upper left corner)

Also the normal state of the input can be defined – opened or closed.

Description	Online Value			Modify Value
☐ DDC Alarming: Motor 1 speed 2.0				
⚙ System functions				
▫ Group alarm from fb/mp/pfb	Only these	←	→	
⚙ Feedback				
▫ Digital input	-1	←	→	
▫ Delay	5.0	←	→	
▫ Alarm status	ok			
⚙ Process feedback				
▫ Digital input	0		→	0
▫ Normal input state	opened	←	→	
▫ Delay (Sec)	30.0	←	→	
▫ Alarm status	ALARM			
⚙ Motor protection				
▫ Digital input	1		→	1
▫ Acknowledgement mandatory	No	←	→	No
▫ Normal input state	opened		→	opened
▫ Alarm suppression	for appl. vltg.	←	→	
▫ Alarm status	ok			
⚙ Service switch				
▫ Digital input	1			

All FBoxes normally monitoring a digital output are able to access the hardware input by themselves. If no input should be accessed use -1 – then no hardware address is defined and this option is disabled – so it's always a choice between FBox input and physically input.

If a physically input is defined within the FBox the commissioning would be complicated during testing the inputs (e.g. turn on/off maintenance switch)

Therefore those FBoxes are displaying at FBox output the input state. This information is enclosed into brackets […]. Here you can see always the raw input state. An output containing an additional exclamation mark - ! – like [fb!] indicates that this input state can be used also to connect to other FBox inputs. E.g. a feed back input is normally always High if motor is running and Low if motor is off.

So it's very easy to define which alarm must be monitored for each motor, you don't have to think during engineering time if there is a feedback, motor protection or maintenance switch input available or not. Just parameterize the input if it's present.

DDC Suite 2.0 / PG5 Building Advanced
Syntax and remarks of actions during workshop

Back to page S01 Start/Stop air condition.

Turn potentiometer to left position until you have the minimum value of 15.0 and 10.0.

This will be the start point to have a defined controller behaviour during the controller functionality explanation.

HDLog
Offline trending

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

To use offline trend functionality in a PCD we have to use the FBox family Historic Data Logging – available since 2 years.

An HDLog Init FBox implements the core functionality, allocating memory and provides an interface to Sweb or ViSi.Plus (or any other SCADA) system.

And then you have to place an FBox for each data you'd like to trend offline in the PCD. This leads often into "trend collecting pages".

Additional you have to
- connect the symbol
- type in a text description
- type in a FBox name property
- set the parameter in adjust window

This means some additional work and you can make also some mistakes, even if you'd like to configure the recording method for all e.g. Set Points to the same style.

HDLog with DDC Suite

Basics

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

If we would like to use the feature Offline trending in PCD within DDC Suite we also have to use the the Historic Data Logging FBox family – that means that the DDC Suite feature is based on the original HDLog FBox functionality!

At least we have to place the FBox HDLog Init – but this is already prepared on first page Initialization; PCD in block COB General.

But you can also see that there is an additional FBox HDLog below the FBox HDLog Init.

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

This new FBox is located in the FBox family DDC Initialization and will only operate with DDC Suite FBoxes.

This FBox has no inputs or outputs, there are only some settings ion adjust window available.

DDC Suite FBoxes are dedicated for Heavac applications and so we normally have some typically data points which are from interest for offline trending. These are Set points, Actual values, Signals and Operating states.

This FBox predefines for each kind of this data type a typically record method, e.g. all Set points should be stored if it's changed with a difference +/- 0.5 – but use also a minimum delay to prevent filling up the offline data base if the set point is changing very fast (e.g. bad calculation). A cyclically recording is not defined (saves memory) and the historic data should be handled in a ring buffer.

Selector	ᵖ X
All	
Filter	

- ⊞ **DDC Controls**
- ⊞ **DDC General**
- ⊟ **DDC Initialisation**
  - Alarm Header 2.0
  - Anti-block protection 2.0
  - BACnet Device 2.0
  - Documentation 2.0
  - Fire dampers 2.0
  - Initialisation 2.0
  - Offline trending 2.0
- ⊞ **DDC Set points**

DDC Initialisation : Offline

Properties

DDC Initialisation:Offline trending 2.0

⊟ **General**	
(Name)	
Comment	
⊟ **Adjust Parameters**	
⊟ **Set points**	
Minimum value difference	0.5
Minimum delay (s)	60
Cyclic record (s)...	0
Type of trend track	Ring storage
⊟ **Actual values**	
Minimum value difference	0.5
Minimum delay (s)	60
Cyclic record (s)...	0
Type of trend track	Ring storage
⊟ **Signals**	
Minimum value difference	2.0
Minimum delay (s)	60
Cyclic record (s)...	0
Type of trend track	Ring storage
⊟ **Operating states**	
Minimum delay (s)	60
Cyclic record (s)...	0
Type of trend track	Ring storage

For this workshop we'll reduce the delay to one second and enable the cyclically recording also with 1 second.

This setting will fill up quite fast the historic data base but for testing I recommend to use this parameters.

1

Properties

DDC Initialisation:Offline trending 2.0

General	
(Name)	
Comment	

Adjust Parameters

Set points

Minimum value difference	0.5
Minimum delay (s)	1
Cyclic record (s)...	1
Type of trend track	Ring storage

Actual values

Minimum value difference	0.5
Minimum delay (s)	1
Cyclic record (s)...	1
Type of trend track	Ring storage

Signals

Minimum value difference	2.0
Minimum delay (s)	1
Cyclic record (s)...	1
Type of trend track	Ring storage

Operating states

Minimum delay (s)	1
Cyclic record (s)...	0
Type of trend track	Ring storage

Static Symbols

Advanced Info

Static Symbols

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

Lets jump in Fupla to block COB System_X and therein to page S01 Start/Stop air condition.

To activate offline trending for DDC Suite FBoxes we don't have to set an additional FBox – all DDC Suite FBoxes supporting offline trending will have an entry in adjust window

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

The DDC Suite FBoxes have always on top in adjust window a group named [--- System functions ---].

Herein the FBox provides different features depending at the functionality of the FBox.

To activate offline trending the parameter PCD Offline Trending (KB)… must be parameterized.

Value 0 disables offline trending in this FBox, any other value reserve the amount of KB you type in. This is the same like in the original HDLog FBoxes.

The FBoxes have of course a lot of adjust parameter and one of them will be used for offline trending.

If you don't know which parameter is used just click at the line
"PCD Offline Trending (KB)…"
and you'll see the parameter to be recorded at the buttom info field.

HDLog with DDC Suite
In use

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

Lets start to activate offline trending for our small air condition application.

On first page we have 2 Sensor FBoxes. Open the adjust windows for the first FBox with name property S01_SupplyAir_Temp.

Lets define 4 KB for this parameter.

S01_SupplyAir_Temp,ref:FuseGuard
Sensor
Inp — Value
AlaHigh
AlaLow

S01_ExhaustAir_Temp,ref:FuseGuard
Sensor
Inp — Value
AlaHigh
AlaLow

Properties

DDC Analogue values:Sensor 2.0

General	
(Name)	S01_Supply_Air_Temp
Reference	FuseGuard
Comment	

Adjust Parameters
System functions

PCD Offline Trending (KB)...	4
PCD Alarm administration (Inde...	0
BACnet	No

Sensor

4

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

Repeat this is with second Sensor FBox. Open adjust windows for first FBox with name property S01_ExhaustAir_Temp.

4

Jump to 3rd page S01 Controller.

Here we have 3 FBoxes.

Open adjust windows of all FBoxes and set parameter
PCD Offline Trending (KB)… to 4.

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

Within this small application we configured 5 offline trend records. Now build your program by pressing key "F2" – you should have no errors.

That's all what we have to do in Fupla. If we would download the program the historic data base would work – but we need of course an application to access the offline trending data.

On the other side if you imagine a big Fupla application with 50 or more pages and up to 300 FBoxes. In Fupla page you don't have any information which FBox is parameterized to use offline trending, how many KB or how you can access the offline data from this FBox.

Therefore the DDC Suite FBoxes creating a text file where all this information are centralized to support you during engineering and also for documentation.

The file name is always DDC_HDLog.txt and is created in the CPU folder.

This file is already available in our CPU001

Workshop [CPU001] - Saia Project Manager SP2.0.1

File   Edit   View   Project   Device   Online   Tools

Project Tree

- Project 'Workshop' ; 2 Devices
- Common Files
- Library Manager
- CPU_Template_EN - PCD3.M5540 - S-Bus Stn 62
- **CPU001 - PCD3.M5540 - S-Bus Stn 62**
  - Online Settings
  - Device Configurator
  - Build Options
  - Program Files
    - .\WEB\SWeb.prj
    - BACnet.bnt
    - DDC_Alarming.CSV
    - DDC_BACnet.bnt
    - DDC_DI_DO_List.CSV
    - DDC_Documentation.htm
    - DDC_HDLog.txt
    - HVC.fup
    - ReadMe.txt
  - Listing Files
  - Documentation Files

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

This file is not linked to the program – this file contains only information about historic trending parameterized with DDC Suite FBoxes.

A double click at this file will open the notepad. You'll see all definitions we made in FBox HDLog and also from each FBox where the parameter PCD Offline Trending (KB)… is set to a value greater than 0 (= activated).

At least each FBox will inform you about
- record type, Set point, actual value …
- The FBox name property
- Symbol to be used in an Sweb trend macro
- effective data point in FBox for trending
- reserved memory size

Don't change the file manually. If you build the program again the file will be overwritten.

CPU001 - PCD3.M5540 - IPNode 99, Station 0
- Settings
- Program Files
  - BACnet.bnt
  - DDC_Alarming.CSV
  - DDC_BACnet.bnt
  - DDC_HDLog.txt
  - HDLog.fun

```
DDC_HDLog.txt - Notepad
File  Edit  Format  View  Help

============================================
PCD offline Trending Settings:
--------------------------------------------
Setpoint :
- min. difference (unit, raw format)      : 5
- minimum delay   (seconds)               : 1
- cyclic delay    (seconds)               : 1
- type            (0=Fill&Stop, 1=Ringbuffer) : 1

Actual value :
- min. difference (unit, raw format)      : 5
- minimum delay   (seconds)               : 1
- cyclic delay    (seconds)               : 1
- type            (0=Fill&Stop, 1=Ringbuffer) : 1

Signal :
- min. difference (unit, raw format)      : 20
- minimum delay   (seconds)               : 1
- cyclic delay    (seconds)               : 1
- type            (0=Fill&Stop, 1=Ringbuffer) : 1

Steuern :
- min. difference (unit, raw format)      : 0
- minimum delay   (seconds)               : 1
- cyclic delay    (seconds)               : 0
- type            (0=Fill&Stop, 1=Ringbuffer) : 1

============================================
Record FBox [Measurement - Sensor]
--------------------------------------------
Type                          : actual value
FBox Properties Name          : S01_SupplyAir_Temp
Use symbol for Sweb           : A.HDLog.S01_SupplyAir_Temp
Effective symbol in record stored : S01.SupplyAir.Temperature.Sensor.Istwert
Used memory                   : 4 KB

============================================
Record FBox [Measurement - Sensor]
--------------------------------------------
Type                          : actual value
FBox Properties Name          : S01_ExhaustAir_Temp
Use symbol for Sweb           : A.HDLog.S01_ExhaustAir_Temp
Effective symbol in record stored : S01.ExhaustAir.Temperature.Sensor.Istwert
Used memory                   : 4 KB

============================================
Record FBox [Regulation - Cooler]
```

At this point we have created a small and nice air condition application. In real life it would be bigger with much more FBoxes or symbols – but we can reuse this in future if we store it as template.

Therefore we export this application as tempalte.
Click on Page in menu and in context menu entry
Export…

Type in file name, use AirCondition_HDLog and
press button Export

Within Dialog Export Pages select option All and finish with button OK.

End of engineering a Fupla application. With DDC Suite FBoxes we reduce the manual work to define symbols for FBoxes – only the symbols in the side connectors must be defined manually.

HDLog with DDC Suite
Accessing data within SWeb

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

We will create a fast Sweb application.

First open your Web Editor Project (Sweb.prj)

Within S-Web Editor open a new page

In dialog New File type in Trending as file name and finish with a click on button Add.

The following steps are standard when using HDLog offline data in a Sweb application.

There is no specific behaviour for DDC Suite!

The new page Trending.teq appears and we have to add a trend macro.

Insert a Macro …

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

Within Select_Insert macro From library dialog

- Select folder SaiaTrendMacro
- then OfflineTrendComplete_5_13_01.esm

And in dialog Configure Macro Infos on macro instance just click on button OK.

After macro import double click in macro.

The Group dialog appears. Activate tab Advanced settings

In list Select the Painter to Configure double click at second
entry 1_MacroOffline Trend ….

In dialog Macro Offline Trend activate tab Macro Offline Trend.

We defined in our Fupla 5 historic trend data, so click 5 times on button Add.

You should see 5 empty entries.

Click on first entry

Then select colour red

Click on parameter Trend's PPO Name button Select to assign a historic data.

In dialog Browse for Symbol [CPU001] to node

- A
- A.HDLog

And select S01_SupplyAir_Temp. You can see in column Comment a reminder where this symbol should be used.

Curves define

	Y Min :	Y Max :	Color :
	0	100	0,0,0
	0	100	0,0,0

Color

Trend's PPO Name

Y Min : 0

Y Max : 100

Select

Update

Browse For Symbol [CPU001]

Group/Symbol △	Type	Address/Value	Comment
A	GROUP		
A.Alarm	GROUP		
A.HDLog	GROUP		
Init	R	2235	"Base PPO Name" in SWeb Trend Macro
S01_Cooler	R	2586	"Trend's PPO Name" in SWeb Trend Macro
S01_ExhaustAir_Temp	R	2584	"Trend's PPO Name" in SWeb Trend Macro
S01_MixedAir	R	2587	"Trend's PPO Name" in SWeb Trend Macro
S01_Preheater	R	2588	"Trend's PPO Name" in SWeb Trend Macro
S01_SupplyAir_Temp	R	2585	"Trend's PPO Name" in SWeb Trend Macro
A.HDLog.Data	GROUP		
S	GROUP		

Select   Cancel

You see the DDC Suite FBox uses the FBox name property to generate automatically a symbol in symbol editor within system tab, group A.HDLog.

Therefore it's necessary to define always the FBox name property – it's also in use for other features …

Set parameter

Y-Min: to 10.0

Y-Max: to 30.0

And click on button Update.

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

Repeat this for all 4 left historic data. Orientate at list below

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

Finally we have to define where the Sweb application will find the data base itself at the PCD.

Therefore click on button Select at parameter Base PPO Name

Select in dialog symbol

A.HDLog.Init

Finally activate checkbox
Automatic Y axes description
and click on button OK. (2 times, because former dialog is still active)

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

Now we have to check some PPO settings, therefore click on button PPO in symbol bar.

You will get a dialog asking for saving changes. Click in button Yes.

You will get a dialog asking for saving project changes. Click in button Yes.

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

You will see the PPOs Initialisation list.

We have to define the format for all symbols containing to S01 air condition.

Set Format to DEC.1

The format for A.HDLog.Init stays unchanged (DEC).

End by clicking on button OK.

Now we have to define a start page for web application.

Click in symbol bar at button HTML and type start.htm into File name text field.

Finish with a click on button Save.

Finally we can build the Sweb application by clicking on button Build All.

Close S-Web Editor.

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

Now add the web server to our CPU.

First add a new program file to your project.

Select from File Type list entry Web Server
Project (*.wsp) and type into text field File Name

WebServer

□ CPU001 - PCD3.M5540 - IPNode 99, Station 0
├ Settings
├ Program Fi...   New...   Ctrl+N
│   .\WEB\   Add Files...
│   BACne
│   DDC_A   Paste   Ctrl+V
│   DDC_B   Delete All Files...
│   DDC_H

New File [CPU001]

File Name:
WebServer

Directory:
D:\1_Projects_PG5_14\Workshop\CPU001

File Type:
Watch Window Files (*.5ww)
Data Transfer Files (*.dt5)
BuES SQL Database Files (*.bue)
BACnet Files (*.bnt)
HMI Files (*.hmi)
Web Server Project (*.wsp)
Web Editor Project (*.prj)

Description:

☑ Linked/Built   ☑ Open file now

Help   OK   Cancel

DDC Suite 2.0 / PG5 Building Advanced
HDLog – offline trending

Select within SAIA – Web-Builder-C all files and add then to list WEB Server Content.

Press button Generate, close Web Builder and build in PG5 the CPU, download .

SWeb alarming

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

To use SWeb alarming functionality in a PCD we have to use the FBox family Alarming – available since 2 years.

An Alarm SWeb FBox implements the core functionality, allocating memory and provides an interface to Sweb or via CGI calls.

And then you have to place some FBoxes to collect all the alarm information in you application. This leads often into "alarm collecting pages".

Additional you have to
- connect the symbol
- type in an alarm index number
- define the alarms by your own
- edit the text in SWeb application CSV file in right order

This means some additional work and you can make also some mistakes, even if you'd like to use more alarm list or more air condition within one alarm list

Alarming with DDC Suite

Basics

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

If we would like to use the feature PCD managed alarming within DDC Suite we also have to use the the Alarming FBox family – that means that the DDC Suite feature is based on the original Alarming functionality!

At least we have to place the FBox Alarm SWeb – but this is already prepared on first page Initialization; PCD in block COB General.

The sub-FBoxes for alarming are connected to this alarm list via FBox Name/Ref property – it's possible (depending on PCD type) to have more alarm list in one PCD.

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

Lets start to define the alarms for SWeb for the air condition.

Activate page S01 Start/Stop air condition in block COB System_X.

To activate alarm management in DDC Suite FBoxes we don't have to set additional FBoxes – all DDC Suite FBoxes supporting alarm management will have an entry in adjust window

The DDC Suite FBoxes have always on top in adjust window a group named [--- System functions ---].

Herein the FBox provides different features depending at the functionality of the FBox.

To activate alarm management the parameter PCD Alarm administration (Index)… must be parameterized.

Value 0 disables alarm managemt in this FBox, any other value defines the base alarm index for the first alarm in this FBox. This is the same like in the original Alarm FBoxes.

Properties	⊄ ✕
DDC Analogue values:Sensor 2.0	

General	
(Name)	S01_Supply_Air_Temp
Reference	FuseGuard
Comment	

Adjust Parameters	
System functions	
PCD Offline Trending (KB)…	4
PCD Alarm administration (Inde…	0
BACnet	No
Sensor	
Card type	Conversion
Correction	0.0

PCD Alarm administration (Inde…	0

The FBoxes have of course a different number of alarm information.

If you don't know how many alarms are handled within this FBox just click at the text PCD Alarm administration (Index)… and you'll see the parameter to be recorded.

Alarming with DDC Suite
In use

Lets start to activate alarm management for our small air condition application.

On first page we have 2 Sensor FBoxes. Open adjust windows for first FBox with name property S01_SupplyAir_Temp.

Lets define the base index for the first alarm in this FBox. We'll start with number 1. If this FBox has more than 1 alarm than it takes automatically also number 2, 3, 4 and so on until all alarms in this FBox are numbered.

1

We also have to type in in next FBox another base index. It's not allowed that they overlap – otherwise you'll allocate 2 alarms from different FBoxes to 1 alarm in alarm management. This leads into strange behaviour.

Therefore it's necessary to know how many alarms this Sensor FBox handles. Just click on parameter PCD Alarm administration (Index)… und you'll see in a pop up window that this FBox handles 2 alarms.

That means:

- We defined within this FBox the index with 1

- 2 alarms are handled

Base index to use in next FBox is at least 3
(This base index + number of alarms)

Open adjust windows for first FBox with name property S01_ExhaustAir_Temp.

The base index we calculated for next FBox is 3, type in in parameter PCD Alarm administration (Index)…

Now we can check again how many alarms this FBox handles. Again 2 – so base index for next FBox is calculated 3+2 = 5

On next page all FBoxes providing alarm(s). Type in the index in parameter PCD Alarm administration (Index)… and calculate the index for next FBox.

Start in upper right corner and follow the arrows to the next FBox.

At least the index should be:
- FBox Motor 1 (S01_SupplyAir_Fan): 5
- FBox AlaMotor1 (S01_SupplyAir_FanAla): 6
- FBox Motor 1 (S01_EchaustAir_Fan): 11
- FBox AlaMotor1 (S01_ExhaustAir_FanAla): 12

At the end we should have total 16 alarms.

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

So far – so good. But if we have more than 1 alarm list – how do the DDC Suite FBoxes know to which alarm list the alarm have to be assigned?

The FBoxes from FBox family Alarming are using the Name/Ref mechanism to assign an FBox to an alarm list – but the DDC Suite FBoxes are using already the FBox property Ref. So we must have another possibility.

Therefore we in DDC Suite FBox family DDC Initialisation the FBox Alarm Header 2.0.

Place this FBox on first page in the Upper left corner.

You see that this FBox provides the FBox property name and it's already predefined to a default alarm list.

Rename the ref ThisAlarmList into GeneralAlarmList

So this FBox is assigned to the alarm list with FBox property name GeneralAlarmList. But not only this FBox – automatically all FBoxes from DDC Suite placed after this FBox also know that their alarms have to be assigned to this alarm list!

You can place this FBox as often as it's necessary, e.g. on every page or only once if you use 1 alarm list in you CPU.

FBox Properties

Name:

Reference:
GeneralAlarmList

Comment:

Help   OK   Cancel

S01.Syste

ref: GeneralAlarmList
Alarm Hdr

So we defined all alarm numbers and also the alarm list. Build the program. Now we could start to implement the Alarm macro in Sweb application – but I don't know which alarm in a FBox is assigned as second or third alarm and also I cannot see any Information in Fupla itself or in symbol editor.

So this means the alarms are "hidden" – not really useful.

But the DDC Suite FBoxes are creating during build process automatically a file with name DDC_Alarming.csv.

This file is already listed in CPU Program Files folder.

Double click on it, Excel should open this file.

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

The CSV file contains 4 columns

A: to which alarm list the alarm is assigned

B: alarm number in alarm administration

C: same as B but with prefix "Alarm_"

D: alarm text

Microsoft Excel - DDC_Alarming.CSV

Datei Bearbeiten Ansicht Einfügen Format Extras Daten Fenster ?

A1   ƒx ListDefinition=1

	A	B	C	D
1	ListDefinition=1	GeneralAlarmList		
2	List_1	1	Alarm_1	FuseGuard 230 VAC missing
3	List_1	2	Alarm_2	FuseGuard 24 VAC missing
4	List_1	3	Alarm_3	FuseGuard 24 VDC missing
5	List_1	4	Alarm_4	FuseGuard phase missing
6	List_1	5	Alarm_5	FuseGuard control voltage
7	List_1	1	Alarm_1	S01_SupplyAir_Temp limit high
8	List_1	2	Alarm_2	S01_SupplyAir_Temp limit low
9	List_1	3	Alarm_3	S01_ExhaustAir_Temp limit high
10	List_1	4	Alarm_4	S01_ExhaustAir_Temp limit low
11	List_1	6	Alarm_6	S01_SupplyAir_FanAla no feedback
12	List_1	7	Alarm_7	S01_SupplyAir_FanAla motor protection
13	List_1	8	Alarm_8	S01_SupplyAir_FanAla maintenance switch
14	List_1	9	Alarm_9	S01_SupplyAir_FanAla no process feedback
15	List_1	10	Alarm_10	S01_SupplyAir_FanAla manual override
16	List_1	5	Alarm_5	S01_SupplyAir_Fan Service
17	List_1	12	Alarm_12	S01_ExhaustAir_FanAla no feedback
18	List_1	13	Alarm_13	S01_ExhaustAir_FanAla motor protection
19	List_1	14	Alarm_14	S01_ExhaustAir_FanAla maintenance switch
20	List_1	15	Alarm_15	S01_ExhaustAir_FanAla no process feedback
21	List_1	16	Alarm_16	S01_ExhaustAir_FanAla manual override
22	List_1	11	Alarm_11	S01_ExhaustAir_Fan Service
23				

When we check the alarm number we'll see that the numbers 1 to 5 are used twice, once for some kind of FuseGuard monitoring and once from our definition.

1	Alarm_1	FuseGuard 230 VAC missing
2	Alarm_2	FuseGuard 24 VAC missing
3	Alarm_3	FuseGuard 24 VDC missing
4	Alarm_4	FuseGuard phase missing
5	Alarm_5	FuseGuard control voltage
1	Alarm_1	S01_SupplyAir_Temp limit high
2	Alarm_2	S01_SupplyAir_Temp limit low
3	Alarm_3	S01_ExhaustAir_Temp limit high
4	Alarm_4	S01_ExhaustAir_Temp limit low

Reason: A DDC Suite template already has 2 default pages with some init FBoxes and already the FBox FuseGuard and Alarm Hdr.

By default they are assigned to alarm list with name GeneralAlarmList and first alarm number 1.

So we have to re-address the alarms – but this is done very fast and quite smart. In CSV file we see that 1 to 5 is used twice – but maybe it would be better to reserve 6 to 10 and start with alarm number 11 in our air condition application.

1. Close Excel
2. Jump to first page S01 Start/Stop air condition
3. open adjust window from FBox Alarm Hdr
4. modify parameter Base alarm index from 1 to 11
5. Close adjust parameter
6. Build program (use key "F2")
7. open file DDC_Alarming.csv in PG5 project manager

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

This looks much better – and it's very easy to "move" alarms numbers for a complete application (e.g. the air condition) instead of renumbering within each FBox.

Lets have a look at column C. The alarm text is created automatically. The FBox using this method:

Use FBox property name and add a detailed information – because one FBox may have more than one alarm.

So the first part can be defined by yourself.

(To modify detailed information see chapter Sweb alarming - advanced)

	C	D
rmList		
1	Alarm_1	FuseGuard 230 VAC missing
2	Alarm_2	FuseGuard 24 VAC missing
3	Alarm_3	FuseGuard 24 VDC missing
4	Alarm_4	FuseGuard phase missing
5	Alarm_5	FuseGuard control voltage
11	Alarm_11	S01_SupplyAir_Temp limit high
12	Alarm_12	S01_SupplyAir_Temp limit low
13	Alarm_13	S01_ExhaustAir_Temp limit high
14	Alarm_14	S01_ExhaustAir_Temp limit low
16	Alarm_16	S01_SupplyAir_FanAla no feedback
17	Alarm_17	S01_SupplyAir_FanAla motor protection
18	Alarm_18	S01_SupplyAir_FanAla maintenance switch
19	Alarm_19	S01_SupplyAir_FanAla no process feedback
20	Alarm_20	S01_SupplyAir_FanAla manual override
15	Alarm_15	S01_SupplyAir_Fan Service
22	Alarm_22	S01_ExhaustAir_FanAla no feedback
23	Alarm_23	S01_ExhaustAir_FanAla motor protection
24	Alarm_24	S01_ExhaustAir_FanAla maintenance switch
25	Alarm_25	S01_ExhaustAir_FanAla no process feedback
26	Alarm_26	S01_ExhaustAir_FanAla manual override
21	Alarm_21	S01_ExhaustAir_Fan Service

At this point we have created a small and nice air condition application. In real life it would be bigger with much more FBoxes or symbols – but we can reuse this in future if we store it as template.

Therefore we export this application as tempalte.
Click on Page in menu and in context menu entry
Export…

Type in file name, use
AirCondition_HDLog_Alarm and press button
Export

Within Dialog Export Pages select option All and finish with button OK.

End of engineering a Fupla application. With DDC Suite FBoxes we reduce the manual work to define symbols for FBoxes – only the symbols in the side connectors must be defined manually.

Using alarm text in Sweb application

OK – back to SWeb engineering. Now we have a CSV file with all necessary information.

Open S-Web Editor

Create a new file.

Alarming

We have to load an alarming macro. Click on Library in menu bar. In context menu select Get object from library …

Select folder SaiaAlarmingMacro

Select macro
MacroAlarmingDefOnline_5_14_05.esm

And click at button OK on input dialog.

After macro import double click in macro.

The Group dialog appears. Activate tab Advanced settings

In list Select the Painter to Configure double click at first
entry 0_Macro DefOnlineAlarm ….

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

In dialog Macro DefOnlineAlarm activate tab Macro Alarming.

First we have to define which alarm list we'd like to show in this view. Therefore click on button Select at parameter Alarm List PPO Name.

Select from group

- System
- System.A
- System.A.Alarm
- System.A.Alarm.GeneralAlarmList

Entry MyHandle end click on button Select.

At least we have to define that the alarm text should be used from a CSV file. Activate checkbox Alarms Text From CSV File.

Finally define how many Alarms in this list should be supported. Type in 50

Close all dialog.

We have to create the CSV file by clicking on button CSV in symbol bar.

In dialog Select type in text field FileName

SWebText

And click on button Open.

Click on button "Yes" on sub dialog.

You should have the CSV file listed in project file list.

Select form menu Project and in context menu entry
Project configurations …

Click on tab Project – Applet Advanced

Activate checkbox HTML Parameters in CSV file

Select in dropdown list csv file: entry SWebText.csv

Close dialog by clicking on button OK.

Open Trending.teq file

Add a button in lower left corner

Set on tab Repaints the parameter Name to Alarming

Activate on tab Actions Jump the checkbox View Jump and select from drop down list entry Alarming.teq

Close dialog.

DDC Suite 2.0 / PG5 Building Advanced
SWeb Alarming

To create the default Alarm text's
we have to press the [icon] button or
choose the command
"Generate HTMLTAGS CSV File"
from the project menu.

Alarming with DDC Suite
AddOn tool

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

The new DDC Suite Addon Tool is automatically installed with PG5 2.0.200

The DDC Suite Addon Tool merges the Alarm text's from the DDC_Alarming.csv into the SWebText.csv file, that is used by the Web Alarming Macro.

DDC Suite 2.0 / PG5 Building Advanced

SWeb alarming

In the project, right click on "Program Files"
Select the "File Type" DDC Suite (*.ddc):

Choose a nice name i.e. DDC_Addon.ddc

Take care that the Option "Linked/Built" has to be deactivated.

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

The Addon tool recognizes the path to the CPU folder and writes per default into the SWebText.csv file.

You can also choose a different output file by clicking the "Browse" button (if you did not use a DDC Suite template).

Click on button Generate.

If the update has been successful a Success text appears in the status bar

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

When you check the **SWebText.scv** file you'll see that the alarm text has been updated completely.

Undefined alarms in DDC_Alarming.csv are removed and replaced with a "-".

The AddOn tool will walk through all alarms defined in the SWebText.csv file and replace unused alarm with a "-"! If you type in manually a text in an unused alarm you'll loos this information!

```
SWebText.csv
A.Alarm.GeneralAlarmList.MyName_31;-
A.Alarm.GeneralAlarmList.MyName_30;-
A.Alarm.GeneralAlarmList.MyName_29;-
A.Alarm.GeneralAlarmList.MyName_28;-
A.Alarm.GeneralAlarmList.MyName_27;-
A.Alarm.GeneralAlarmList.MyName_26;-
A.Alarm.GeneralAlarmList.MyName_25;S01_ExhaustAir_FanAla manual override
A.Alarm.GeneralAlarmList.MyName_24;S01_ExhaustAir_FanAla no process feedback
A.Alarm.GeneralAlarmList.MyName_23;S01_ExhaustAir_FanAla maintenance switch
A.Alarm.GeneralAlarmList.MyName_22;S01_ExhaustAir_FanAla motor protection
A.Alarm.GeneralAlarmList.MyName_21;S01_ExhaustAir_FanAla no feedback
A.Alarm.GeneralAlarmList.MyName_20;S01_ExhaustAir_Fan Service
A.Alarm.GeneralAlarmList.MyName_19;S01_SupplyAir_FanAla manual override
A.Alarm.GeneralAlarmList.MyName_18;S01_SupplyAir_FanAla no process feedback
A.Alarm.GeneralAlarmList.MyName_17;S01_SupplyAir_FanAla maintenance switch
A.Alarm.GeneralAlarmList.MyName_16;S01_SupplyAir_FanAla motor protection
A.Alarm.GeneralAlarmList.MyName_15;S01_SupplyAir_FanAla no feedback
A.Alarm.GeneralAlarmList.MyName_14;S01_SupplyAir_Fan Service
A.Alarm.GeneralAlarmList.MyName_13;S01_ExhaustAir_Temp limit low
A.Alarm.GeneralAlarmList.MyName_12;S01_ExhaustAir_Temp limit high
A.Alarm.GeneralAlarmList.MyName_11;S01_SupplyAir_Temp limit low
A.Alarm.GeneralAlarmList.MyName_10;S01_SupplyAir_Temp limit high
A.Alarm.GeneralAlarmList.MyName_9;-
A.Alarm.GeneralAlarmList.MyName_8;-
A.Alarm.GeneralAlarmList.MyName_7;-
A.Alarm.GeneralAlarmList.MyName_6;-
A.Alarm.GeneralAlarmList.MyName_5;FuseGuard control voltage
A.Alarm.GeneralAlarmList.MyName_4;FuseGuard phase missing
A.Alarm.GeneralAlarmList.MyName_3;FuseGuard 24 VDC missing
A.Alarm.GeneralAlarmList.MyName_2;FuseGuard 24 VAC missing
A.Alarm.GeneralAlarmList.MyName_1;FuseGuard 230 VAC missing
Select Mode :;Select Mode :
Sort Mode :;Sort Mode :
```

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

Back in S-Web Editor.

- save the project by clicking on disk symbol button
- Build the S-Web project by clicking on build button

Close S-Web Editor.

Open WebServer project from PG5 project manager

Select from left list
- Alarming.teq
- SWebText.csv

Click on button Add, press Generate
End WebServer.

Build program in PG5 project manager
Download program into PCD

SWeb alarming - advanced

Remember – the alarm text is created by

- FBox property Name

- detailed information – because one FBox may have more than one alarm

So the first part can be defined by yourself – but the second part is "hard coded in FBox", e.g. the FBox "Sensor" from family "Analogue values" supports 2 alarm, "limit low" and "limit high".

How can this default texts customized?

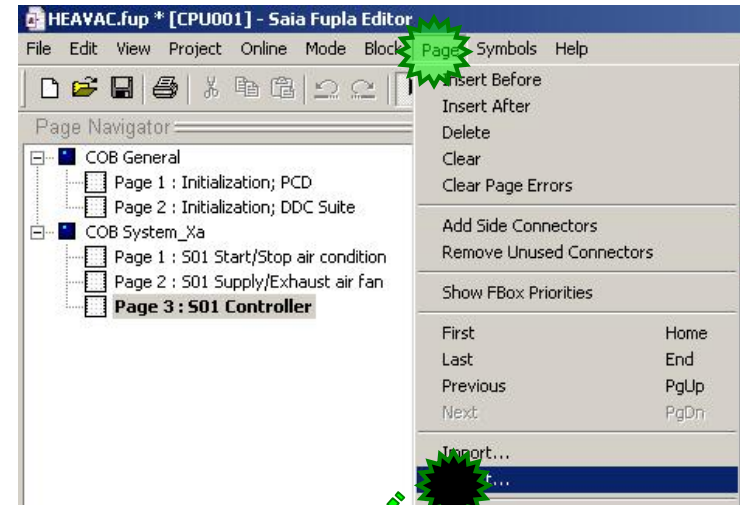|	C	D
rmList		
1	Alarm_1	FuseGuard 230 VAC missing
2	Alarm_2	FuseGuard 24 VAC missing
3	Alarm_3	FuseGuard 24 VDC missing
4	Alarm_4	FuseGuard phase missing
5	Alarm_5	FuseGuard control voltage
11	Alarm_11	S01_SupplyAir_Temp limit high
12	Alarm_12	S01_SupplyAir_Temp limit low
13	Alarm_13	S01_ExhaustAir_Temp limit high
14	Alarm_14	S01_ExhaustAir_Temp limit low
16	Alarm_16	S01_SupplyAir_FanAla no feedback
17	Alarm_17	S01_SupplyAir_FanAla motor protection
18	Alarm_18	S01_SupplyAir_FanAla maintenance switch
19	Alarm_19	S01_SupplyAir_FanAla no process feedback
20	Alarm_20	S01_SupplyAir_FanAla manual override
15	Alarm_15	S01_SupplyAir_Fan Service
22	Alarm_22	S01_ExhaustAir_FanAla no feedback
23	Alarm_23	S01_ExhaustAir_FanAla motor protection
24	Alarm_24	S01_ExhaustAir_FanAla maintenance switch
25	Alarm_25	S01_ExhaustAir_FanAla no process feedback
26	Alarm_26	S01_ExhaustAir_FanAla manual override
21	Alarm_21	S01_ExhaustAir_Fan Service

Each FBox generating alarm text supports an external file.

- If the file is not present than the "hard coded" detailed information is used.
- if the file is found then the definition how to create the alarm must be declared in this file and the "hard coded" part is ignored.

So the external file disables the "hard coded" definition!

The files are structured with a strong name convention
- ALM_ - declares that this file contains alarm declaration
- DDC_ - identifies that this file is used with DDC suite FBoxes
- "Family_" e.g. Alarming_ - declares the DDC Suite family
- "FBox" e.g. 1Alarm – defines the FBox itself
- .src – file extenstion

Example: ALM_DDC_Alarming_1Alarm.src

But you don't have to know all these file names or create them by yourself.

Within each DDC Suite template this files are already existing in the template CPU folder.

So in fact the "hard coded" definition is not used by default – but the files containing the "hard coded" part itself.

We just prepared the files that you don't have to create them by your own or copy them from other folder into CPU folder.

If you made some mistakes with this files – you can restore them.

Within a DDC Suite project you'll find in folder FBox_AddOns, subfolder SWeb Alarming all files.

Just copy them and past it into your CPU folder.

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

Now lets customize the detailed alarm information for FBox Sensor.

The FBox Sensor is from FBox family Analogue Values. Lets check if there is a file named

ALM_DDC_AnalogueValue_Sensor.src

Unfortunately not – but when we check the files you'll find a file named

ALM_DDC_Measurement_Sensor.src

During translating DDC Suite into English we hade some different texts within FBoxes, files and description – there may be some small differences – but at the end you can identify them normally.

At least – there is a chapter DDC Suite - advanced – detailed information wherein you'll find for each FBox if it supports an external files for alarming and the file name. Please refer this chapter if you are not able to find the file you're looking for.

Please open file ALM_DDC_Measurement_Sensor.src with Notepad.

You'll see for each alarm an entry like $WRFILE "DDC_Alarming.CSV" …. Always terminated with a construct of @&Name@. This indicates "use here the FBox property Name".

The text after this construct is the detailed information and can be changed without problems.

Please modify "limit high" into "broken wire" and "limit low" into "short circuit". Save file and close Notepad.

DDC Suite 2.0 / PG5 Building Advanced
SWeb alarming

Close Excel and build program.

A "Rebuild All Files" is required – we didn't change any file listed in program files folder within PG5 project manager – so we have to force to rebuild the while program to be sure that the changes in such external files are executed.

Open file DDC_Alarming.csv in PG5 project manager.

You see it's very easy to customize the default alarm texts e.g. if it's a demand from end user.

The FBoxes will search for external files
1st in CPU folder, if there not present
2nd in PG5 libs folder, if there not found
3rd "hard coded" definition is used

	A	B	C	D	E	F	G
3	List_1	2	Alarm_2	FuseGuard 24 VAC missing			
4	List_1	3	Alarm_3	FuseGuard 24 VDC missing			
5	List_1	4	Alarm_4	FuseGuard phase missing			
6	List_1	5	Alarm_5	FuseGuard control voltage			
7	List_1	11	Alarm_11	AC01_Shop_SupplyAir_Temp broken wire			
8	List_1	12	Alarm_12	AC01_Shop_SupplyAir_Temp short circuit			
9	List_1	13	Alarm_13	AC01_Shop_ExhaustAir_Temp broken wire			
10	List_1	14	Alarm_14	AC01_Shop_ExhaustAir_Temp short circuit			
11	List_1	16	Alarm_16	AC01_Shop_SupplyAir_FanAla no feedback			
12	List_1	17	Alarm_17	AC01_Shop_SupplyAir_FanAla motor protection			
13	List_1	18	Alarm_18	AC01_Shop_SupplyAir_FanAla maintenance switch			
14	List_1	19	Alarm_19	AC01_Shop_SupplyAir_FanAla no process feedback			
15	List_1	20	Alarm_20	AC01_Shop_SupplyAir_FanAla manual override			
16	List_1	15	Alarm_15	AC01_Shop_SupplyAir_Fan Service			
17	List_1	22	Alarm_22	AC01_Shop_ExhaustAir_FanAla no feedback			
18	List_1	23	Alarm_23	AC01_Shop_ExhaustAir_FanAla motor protection			
19	List_1	24	Alarm_24	AC01_Shop_ExhaustAir_FanAla maintenance switch			
20	List_1	25	Alarm_25	AC01_Shop_ExhaustAir_FanAla no process feedback			
21	List_1	26	Alarm_26	AC01_Shop_ExhaustAir_FanAla manual override			
22	List_1	21	Alarm_21	AC01_Shop_ExhaustAir_Fan Service			
23	List_1	31	Alarm_31	HC01_Outdoor_Ttemp broken wire			
24	List_1	32	Alarm_32	HC01_Outdoor_Ttemp short circuit			
25	List_1	35	Alarm_35	HC01_Inflow_Temp broken wire			
26	List_1	36	Alarm_36	HC01_Inflow_Temp short circuit			
27	List_1	33	Alarm_33	HC01_Inflow_Temp_Tolerance limit high			
28	List_1	34	Alarm_34	HC01_Inflow_Temp_Tolerance limit low			
29	List_1	37	Alarm_37	HC01_Returnflow_Temp broken wire			
30	List_1	38	Alarm_38	HC01_Returnflow_Temp short circuit			

BACnet

BACnet engineering is done with the PG5 BACnet Configurator.

This tool supports you to generate BACnet objects and mapping PCD resources to BACnet objects.

At least the engineer must know which resource should be mapped within which BACnet objects, and if done this way what must I do in Fupla?

BACnet is not transport protocol like S-Bus or ModBus. It's a functionality and must be known very well from the engineer. But this takes a long time to learn and getting specialized to BACnet.

DDC Suite will do this part – you don't have to "fight" with BACnet basics.

saia-burgess
Control Systems and Components

BACnet with DDC Suite

In use

In first page (Initialization; PCD) the FBox BACnet is already present. This FBox is located in FBox family DDC Initialization and must be placed only once in a program.

The FBox property name is used to define the BACnet device name.

Modify the name property from DeviceName into Workshop_BACnet.

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Open adjust window. This FBox provides only two parameters.

Parameter Activate BACnet

- If No is selected than no program does not need a BACnet stack running on PCD – even if maybe in some FBoxes BACnet functionality is selected
- If Yes is selected than BACnet stack must run on PCD – otherwise some FBoxes parameterized with BACnet functionality won't work!

Parameter Activate BACnet

Is a unique ID in BACnet network. Set Device ID to

3280

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Jump to page S01 Start/Stop air condition.

Now we have to walk through all FBoxes and parameterize if this FBox should support BACnet and which parameter.

Open adjust window of FBox System 1sp. In BACnet parameters drop down list just select if only the HMI (switch) should be activated for BACnet or also the clock should be handled by BACnet.

Select HMI/Clock and close adjust window.

Properties

DDC Systems and Clocks:System 1 speed 2.0

General	
(Name)	S01_Switch
Comment	

Adjust Parameters	
System functions	
BACnet	**HMI/clock**
Settings	
HMI Low prio	Off
... Clock accessed by	Input
... calendar channel	Not used

HMI Low prio

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Next page, first Motor 1 FBox
(S01_SupplyAir_Fan)

first AlaMotor1 FBox
(S01_SupplyAir_FanAla)

2nd Motor 1 FBox
(S01_ExhaustAir_F
an)

2nd AlaMotor1 FBox
(S01_ExhaustAir_F
anAla)

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Next page, FBox Val
(S01_SupplyAirTempSetPoint)

FBox Cooler (S01_Cooler)

FBox MixedAir (S01_MixedAir)

And finally FBox PreHeater (S01_Preheater)

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Build program. During build DDC Suite FBoxes creating all BACnet objects and mapping for the selected BACnet functionalities.

This file is called DDC_BACnet.bnt – and it's a fully parameterized BACnet configuration. If there is no need to change something, e.g. like unit or scale, the file can be used immediately to be linked to the program and downloaded.

But first let's have a look into this file.

DDC Suite 2.0 / PG5 Building Advanced
BACnet

You see that there is a device "Workshop_BACnet" with ID 3280.

We also see that there are 59 BACnet objects. We took 2 Minutes to define all this BACnet objects in Fupla – simply by selecting functionality in FBox.

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Select in object list S01_ExhaustAir_Temp Analog [AI 1]

And you get all properties.

As you can see there are a lot of PCD resources mapped into this object – at least the property Present Value is mapped.

Depending on functionality more than 1 resource has been implemented. So this is real BACnet functionality and not only providing the temperature value for BACnet.

In this example e.g. also the limit low/high and unit low/high are implemented.

Name	Value/Link
Present Value	%(S01.ExhaustAir.Temperature.Sensor.Istwert)
Description	S01_ExhaustAir_Temp / S01.ExhaustAir.Temperature.S
Device Type	---
Reliability	no-fault-detected
Out Of Service	FALSE
Update Interval	1
Units	degrees-Celsius
Min Pres Value	%(S01.ExhaustAir.Temperature.Sensor.IstwertY1)
Max Pres Value	%(S01.ExhaustAir.Temperature.Sensor.IstwertY2)
Resolution	0.1
COV Increment	1
Time Delay	0
Notification Class	0
High Limit	%(S01.ExhaustAir.Temperature.Sensor.GwOben)
Low Limit	%(S01.ExhaustAir.Temperature.Sensor.GwUnten)
Deadband	2
Limit Enable	(1,1)
Event Enable	(1,1,1)
Notify Type	alarm
Profile Name	---
Unsolicited COV Enabled	FALSE
Event Message Text	---

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Example FBox Sensor

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Example FBox Motor 1 (Commandable object)

DDC Suite 2.0 / PG5 Building Advanced
BACnet

As you see in object list the BACnet object names are created automatically using this method:

Use FBox property name and add a detailed information – because one FBox may have more than one BACnet objects.

So the first part can be defined by yourself

At this point we have created a small and nice air condition application. In real life it would be bigger with much more FBoxes or symbols – but we can reuse this in future if we store it as template.

Therefore we export this application as tempalte.
Click on Page in menu and in context menu entry
Export…

Type in file name, use
AirCondition_HDLog_Alarm_BACnet and press
button Export

Within Dialog Export Pages select option All and finish with button OK.

End of engineering a Fupla application. With DDC Suite FBoxes we reduce the manual work to define symbols for FBoxes – only the symbols in the side connectors must be defined manually.

BACnet with DDC Suite
AddOn tool

DDC Suite 2.0 / PG5 Building Advanced
BACnet

The BACnet configuration DDC_BACnet.bnt created from the DDC Suite FBoxes normally does not fit perfectly, e.g. if you use a Sensor FBox for a pressure sensor.

All parameters are mapped in the right way – but there's no parameter to select the unit in the FBox – and therefore the BACnet object is defined by default with unit °C.

This can be changed manually very easy and it's not a big thing – but the file DDC_BACnet.bnt will be created during the build and all changes done manually in this file will be lost.

On the other side the object ID (a number from 0 … ??) must be always the same for an object, e.g. the outdoor temperature once created as AI 67 must be always AI 67 – because other BACnet client may use this object to get the outside temperature from our PCD.

But also this ID may be changed if the program is build and a new FBox has been placed – because the DDC Suite will always start from ID 0 for all objects.

To fix this problem the BACnet add on tool must be used.

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Let's prepare the Fupla for a clarification how the AddOn tool will work in detail.

Jump to Page 1 of air condition

And disable in FBox Sensor for supply air temperature then BACnet option. Select No

And Build the program.

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Within PG5 project manager / CPU001 we already have a BACnet.bnt configuration. This contains only the device and a notification class and should remember you to link this file to the program and not the file DDC_BACnet.bnt.

When the PG5 project has been build, you can find the DDC_BACnet.bnt file in the device folder. Also visible under "Program Files" in the PG5 project tree.

Start the DDC Suite Addon Tool by clicking the *.ddc file from the Program Files.

Note: When you use the BACnet Addon for the first time the option „Force Update" must be activated.

Press the "Generate" button.
If no MessageBox occurs and the update, creation has been successful. The updates are shown in a log file.

If you use the BACnet Addon later on the „Force Update" option sholud be always deactivated to avoid overwriting the content of the BACnet.bnt file.

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Open BACnet configurator to check the settings.

For better understanding we have to sort the objects by type.

Click in context menu on entry "View" and then "Sort by Type"

DDC Suite 2.0 / PG5 Building Advanced
BACnet

You'll see that object AI 0 (analogue input ID 0) is defined from Sensor FBox with name property

S01_ExhaustAir_Temp

because this was the first FBox generating an analogue input object.

Maybe now a client will be parameterized to read AI 0 from device 3280 to get the exhaust air temperature.

If we change the ID of an object than also all clients must be updated! Therefore it's absolutely necessary to keep them unchanged!

Close BACnet configurator and turn on in Sensor FBox of supply air temperature the BACnet option to Yes

And build the program.

Open file DDC_BACnet.bnt – and you'll see that within this configuration the exhaust air temp. object now is ID 1 – because the Sensor FBox for supply air temp. is in front of the exhaust air temp.

So after a build the object ID in DDC_BACnet.bnt file can be always different – depending on the settings or if you removed/added an FBox!

DDC Suite 2.0 / PG5 Building Advanced
BACnet

OK – but we use the BACnet.bnt file and this file is not changed from DDC Suite. So exhaust air temp. object is still 0 – but we have to add the new supply air temp. object.

But this object also is defined with ID 0 – and two objects of same type cannot have the same ID.

How to solve this problem? The AddOn will do this for you.

Start the DDC AddOn and press the generate button.

The tool generates an Log file that lists all bacnet objects that have been added to the bacnet.bnt file

The tool checks the last used object ID – in this case ID 0 was the last AI ID used – and copies the new object into BACnet.bnt file – updating the original ID (which is in fact not relevant) to the next free ID.

So we have moved the new object into BACnet.bnt without changing ID enumeration of existing objects.

DDC Suite 2.0 / PG5 Building Advanced
BACnet

When now checking the BACnet.bnt file you'll see that the exhaust air temp. object is still AI 0 and supply air temp. object has become AI 1.

To check if an object is already existing in BACnet.bnt file the BACnet AddOn tool must be sure that it is handling the objects in a proper way. You can e.g. change in BACnet.bnt file manually

- Object ID
- Object name
- Description
- or any other property

So how can the tool find out that e.g. object "AI 27 SupplyAirTemp" in BACnet.bnt file is the same as "AI13 AnyName" ?

The tool is checking the property "Present Value" of all objects. The BACne configurator accepts a symbol reference only once in a whole project. So if the symbol is the same – the tool expects that it is the same object – even if everything else is different!

Name	Value/Link
Present Value	%(S01.ExhaustAir.Temperaturer.Sensor.PhysVal)
Description	S01_ExhaustAir_Temp / S01.ExhaustAir.Temperaturer.Sensor.F
Device Type	...

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Therefore:

- never change manually file DDC_BACnet.bnt – you'll loose all changes after a build
- never link file DDC_BACnet.bnt to program – the object ID's may change and a client will get wrong data

- Always use file BACnet.bnt to be linked to program
- update file after a compile to add new BACnet objects from DDC Suite FBoxes into file BACnet.bnt
- change objects only in BACnet.bnt file. AddOn tool won't change any property

BACnet – advanced

⚠ **Only use this if you have to modify the creation of BACnet objects by the DDC Suite Fboxes. Advanced knowledge required**

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Remember – the BACnet object is created by

- FBox property Name
- detailed information – because one FBox may have more than one alarm

So the first part can be defined by yourself – but the second part is "hard coded in FBox", e.g. the FBox "Cooler" from family "Controller" supports the PID parameter "P-range", "Integration time" and "Derivation time".

How can this default texts be customized?

Each FBox generating BACnet objects supports an external file.

- If the file is not present than the "hard coded" detailed information is used.
- if the file is found then the definition how to create the alarm must be declared in this file and the "hard coded" part is ignored.

So the external file disables the "hard coded" definition!

The files are structured with a strong name convention
- BAC_ - declares that this file contains BACnet declaration
- DDC_ - identifies that this file is used with DDC suite FBoxes
- "Family_" e.g. Alarming_ - declares the DDC Suite family
- "FBox" e.g. 1Alarm – defines the FBox itself
- .src – file extenstion

Example: BAC_DDC_Alarming_1Alarm.src

But you don't have to know all these file names or create them by yourself.

Within each DDC Suite template this files are already existing.

We just prepared these files to give you the possibility to modify the generation of BACnet objects.

Within a DDC Suite template project you'll find in folder PG5_Templates/FBox_AddOn, subfolder BACnet all files.

Just copy them and paste them into your CPU folder.

Now lets customize the detailed BACnet object information for FBox Cooler.

The FBox Cooler is from FBox family Controller. Lets check if there is a file named

BAC_DDC_Controller_Cooler.src

Unfortunately not – but when we check the files you'll find a file named

BAC_DDC_RegulationCooler.src

During translating DDC Suite into English we hade some different texts within FBoxes, files and description – there may be some small differences – but at the end you can identify them normally.

At least – there is a chapter DDC Suite - advanced – detailed information wherein you'll find for each FBox if it supports an external files for alarming and the file name. Please refer this chapter if you are not able to find the file you're looking for.

DDC Suite 2.0 / PG5 Building Advanced
BACnet

Please open file BAC_DDC_RegulationCooler.src with Notepad.

This files contains informations how the Fbox should generate it´s BACnet Objects.

It's possible to change names.

Screenshot of Notepad showing BAC_DDC_RegulationCooler.src:

```
; ==============================================================
; -------- BACnet - File
; ==============================================================
;
$IF adj_BACnet>0
;
; ---------------- Valve min
;
                    $WRFILE "DDC_BACnet.bnt" [AV @A.BACnet.AV.Index@]
                    $WRFILE "DDC_BACnet.bnt" object-name = @ &Name@ ValveMin
                    $WRFILE "DDC_BACnet.bnt" present-value = %(@&stc_YMin@) || WS(10.00)
                    $WRFILE "DDC_BACnet.bnt" description = Valve min / @ &Name@
                    $WRFILE "DDC_BACnet.bnt" status-flags = (0,0,0,0)
                    $WRFILE "DDC_BACnet.bnt" reliability = no-fault-detected
                    $WRFILE "DDC_BACnet.bnt" out-of-service = FALSE || W
                    $WRFILE "DDC_BACnet.bnt" update-interval = 1
                    $WRFILE "DDC_BACnet.bnt" units = percent
                    $WRFILE "DDC_BACnet.bnt" cov-increment = 1
                    $WRFILE "DDC_BACnet.bnt" unsolicited_cov_enabled = FALSE
                    $WRFILE "DDC_BACnet.bnt"
;
                    A.BACnet.AV.Index DEF A.BACnet.AV.Index +1 ;
```

Now lets modify the names of the PID parameters.

You have to search for the following lines of code in the .bnt file

```
; ----------------- Proportional range
;
                   $WRFILE "DDC_BACnet.bnt" [AV @A.BACnet.AV.Index@]
                   $WRFILE "DDC_BACnet.bnt" object-name = @&Name@ ProportionalRange
                   $WRFILE "DDC_BACnet.bnt" present-value = %(@&stc_P_Band@) || WS(10.00)

; ----------------- Integration time
;
                   $WRFILE "DDC_BACnet.bnt" [AV @A.BACnet.AV.Index@]
                   $WRFILE "DDC_BACnet.bnt" object-name = @&Name@ IntegrationTime
                   $WRFILE "DDC_BACnet.bnt" present-value = %(@&stc_I_Zeit@) || WS(10.00)

; ----------------- Derivation time
;
                   $WRFILE "DDC_BACnet.bnt" [AV @A.BACnet.AV.Index@]
                   $WRFILE "DDC_BACnet.bnt" object-name = @&Name@ DerivationTime
                   $WRFILE "DDC_BACnet.bnt" present-value = %(@&stc_D_Anteil@) || WS(10.00)
```

The Object Name is created with the Fbox Name (@&Name@) plus an extension i.e. Proportional Range

This extension can be modifieed according to your needs

Now lets modify the names of the PID parameters.

Please modify "ProportionalRange" into "P_range", "IntegrationTtime" into "I_time" and "DerivationTime" into "D_time". Save file and close Notepad. Attention: the "text" must follow the symbol convention of PG5 – no special characters.

```
; ---------------- Proportional range
;
                $WRFILE "DDC_BACnet.bnt" [AV @A.BACnet.AV.Index@]
                $WRFILE "DDC_BACnet.bnt" object-name = @&Name@ ProportionalRange        object-name = @&Name@ P_range
                $WRFILE "DDC_BACnet.bnt" present-value = %(@&stc_P_Band@) || WS(10.00)

; ---------------- Integration time
;
                $WRFILE "DDC_BACnet.bnt" [AV @A.BACnet.AV.Index@]
                $WRFILE "DDC_BACnet.bnt" object-name = @&Name@ IntegrationTime          object-name = @&Name@ I_time
                $WRFILE "DDC_BACnet.bnt" present-value = %(@&stc_I_Zeit@) || WS(10.00)

; ---------------- Derivation time
;
                $WRFILE "DDC_BACnet.bnt" [AV @A.BACnet.AV.Index@]
                $WRFILE "DDC_BACnet.bnt" object-name = @&Name@ DerivationTime            object-name = @&Name@ D_time
                $WRFILE "DDC_BACnet.bnt" present-value = %(@&stc_D_Anteil@) || WS(10.00)
```

A "Rebuild All Files" is required – we didn't change any file listed in program files folder within PG5 project manager – so we have to force to rebuild the whole program to be sure that the changes in such external files are executed.

Open file DDC_BACnet.bnt in PG5 project manager.

You see it's very easy to customize the default object texts e.g. if it's a demand from end user.

The FBoxes will search for external files
1st in CPU folder, if there not present
2nd in PG5 libs folder, if there not found
3rd "hard coded" definition is used

DDC_BACnet.bnt [CPU_Template_EN] - BACnet Configurator

Project   Edit   Configuration   View   Help

[AV 33] S1_Cooler_Controller P_range
[AV 34] S1_Cooler_Controller I_time
[AV 35] S1_Cooler_Controller D_time

Using templates

During the last workshops we programmed a nice small air condition with

- 3 Fupla pages, basic functionality
- Offline trending in PCD
- Alarm management in PCD
- BACnet configuration

In real life we expect to use this air condition within another CPU in this project or in another project. Therefore it would be nice if we create a template.

Lets see how easy we can do this.

Creating a new CPU in the project

DDC Suite 2.0 / PG5 Building Advanced
Using templates

We'll start with a new CPU.

Therefore we use the copy/p0aste in project manager. Click with right mouse button on "CPU _Template_EN" and then Copy in context menu.

Click with right mouse button on "Project 'Workshop'" and then Paste CPU in context menu.

We have to rename the CPU, please use "CPU002" and press "OK".

Open HEAVAC.fup from CPU002

Jump in Fupla to first page of COB
System_X

Let's rename the COB, here we'll import
the air condition we designed before.

AirCondition_01

DDC Suite 2.0 / PG5 Building Advanced
Using templates

Lets reuse the template within this CPU. Click in menu bar at Page and in context menu at Import…

We have generated already 4 templates:
-AirCondition
-AirCondition_HDLog
-AirCondition_HDLog_Alarm
-AirCondition_HDLog_Alarm_BACnet

Let's use the complex template. Select file
AirCondition_HDLog_Alarm_BACnet.fxp

1. Uncheck Set Internal Variables to system defined

   This is really necessary – otherwise you'll import the template and all FBoxes won't have internal symbols. But we restructured and renamed the groups to reuse them easily!

2. Select option After current page

DDC Suite 2.0 / PG5 Building Advanced
Using templates

Change in column Page Name the Prefix
S01 into AC01_Shop with Find &
Replace

**DO NOT OK Button! OTHERWISE
YOU WILL START IMPORT!**

Activate tab Symbols List

Right mouse click and select Find and Replace.

DO NOT OK Button! OTHERWISE YOU WILL START IMPORT!

Activate tab FBox List

Click right mouse button and select from
context menu Replace prefix

Enter in text field Find what: S01_

Enter in text field Replace with: AC01_Shop_

Click on button Replace All.

Click on button Cancel to close dialog.

And start import by clicking on button OK.

With this few changes during import we have a complete copy from our air condition, everything renamed into AC01_Shop

DDC Suite 2.0 / PG5 Building Advanced
Using templates

Documentation file DDC_HDLog.txt now also contains historic data information from AC01_Shop

Also alarm file DDC_Alarming.csv now also contains alarm numbers and text information from AC01_Shop

And of course also BACnet file
DDC_BACnet.csv now contains all
BACnet objects from AC01_Shop

Multiple Import

DDC Suite 2.0 / PG5 Building Advanced
Using templates – multiple import

Often we need the same template more than once in our CPU, e.g. a heating circuit will be up to 5 times implemented.

We can import it 5 times and replace always the prefix manually. This is quite fast – but it can be faster. Therefore Fupla provides a "Multiple import" which is based on indexing.

Let's create a new block to import the heating cirquit 5 times.

Type in Heating_Circuit

Lets reuse the template within this CPU. Click in menu bar at Page and in context menu at Import…

The DDC Suite already contains some basic templates. We have to go to the PG5_Templates folder and there select the Fupla_Templates fol…

Then you'll see that the project also contains a folder "Fupla Templates"

DDC Suite 2.0 / PG5 Building Advanced
Using templates – multiple import

Open the folder "Fupla_Templates" with a double click.

We have some templates for systems - heating circuit, air condition - and functionalities – like calendar.

Open folder "Systems" with a double click.

Select from "Systems" folder the "HeatCircuit" folder.

And there we'll import the template

20_HeatCircuit_Inflow_Controlled_1Pump.

1. Uncheck Set Internal Variables to system defined

   This is really necessary – otherwise you'll import the template and all FBoxes won't have internal symbols. But we restructured and renamed the groups to reuse them easily!

2. Select option After current page

DDC Suite 2.0 / PG5 Building Advanced
Using templates – multiple import

We'd like to import the heating circuit template immediately 5 times. Therefore we have to define the parameter "Copies Number" to 5

Change in column Page Name the Prefix HeatCirc_T1 into HC0#
You can also use Find & Replace to do this

The "#" will be replaced with the "Base Index" and incremented x-times depending on "Copies Numbers:"

Select all pages 1..4 in Page Range area

**DO NOT USE OK Button!
OTHERWISE YOU WILL START
IMPORT!**

Import dialog:
General | Symbol List | FBox List | Source

Insert
Number of Copies: 5
Base Index: 1
○ Before current page
● After current page

Advanced
☐ Set Internal Variables to system defined
☐ Reset page conditions

Page Range

No.	Page Name	Description
1	HC0#	General
2	HC0#	Alarm/Report
3	HC0#	Enable
4	HC0#	Regulation/Pump

OK   Cancel   Help

Activate tab Global Symbols

Rename in first row, cell Group entry
HeatCirc_T1 into HC0#

**DO NOT PRESS THE OK BUTTON!
OTHERWISE YOU WILL START
IMPORT!**

You'll see that all symbols are marked red –
indication that all symbols are now
moved into new group HC0#

Activate tab FBox List

Mark all entries, click with right mouse
button and select from context menu
Replace prefix

**DO NOT PRESS THE OK BUTTON!
OTHERWISE YOU WILL START
IMPORT!**

Enter in text field Find what: HeatCirc_T1_

Enter in text field Replace with: HC0#_

Click on button Replace All.

Click on button Cancel to close dialog.

And start import by clicking on button OK.

DDC Suite 2.0 / PG5 Building Advanced
Using templates – multiple import

With this few changes during import
we have 5 times a complete copy
from the heating circuit template
from DDC Suite, everything
renamed into HC01, HC02 .. HC05

Page navigator 〒 ✕

- COB Initilization
  - 1: Initialisation; PCD
  - **2: Initialisation; DDC Suite**
- COB HeatingCircuits
  - 1: HC01; General
  - 2: HC01; Alarm/Report
  - 3: HC01; Enable
  - 4: HC01; Regulation/Pump
  - 5: HC02; General
  - 6: HC02; Alarm/Report
  - 7: HC02; Enable
  - 8: HC02; Regulation/Pump
  - 9: HC03; General
  - 10: HC03; Alarm/Report
  - 11: HC03; Enable
  - 12: HC03; Regulation/Pump
  - 13: HC04; General
  - 14: HC04; Alarm/Report
  - 15: HC04; Enable
  - 16: HC04; Regulation/Pump
  - 17: HC05; General
  - 18: HC05; Alarm/Report
  - 19: HC05; Enable
  - 20: HC05; Regulation/Pump
- COB AirCondition_01
  - 1: AC01_Shop_Start/Stop AirCondition
  - 2: AC01_Shop_Supply/Exhaust Air fan
  - 3: AC01_Shop_Controller

Symbol Editor 〒 ✕

Find:

Symbol Name	Type	Address/Val...	Comment
HVC.fup	ROOT		
PCD	GROUP		
AC01_Shop	GROUP		
Controls	GROUP		
Calendar	GROUP		
HC01	GROUP		
HC02	GROUP		
Inflow	GROUP		
Returnflow	GROUP		
Outdoor	GROUP		
System	GROUP		
HC03	GROUP		
HC04	GROUP		
HC05	GROUP		

All Publics | System | HVC.fup ✕

On YouTube you can find a video, that explains how to do a multiple import :

http://www.youtube.com/watch?v=E0LJsXTtN1Y

ToDo's after an import

After import a "build" should work without errors.

But we have to check all topics of a template, maybe we have to modify some data due to duplicate addresses or definitions. Let's check them step by step:

Application itself. Good structured template systems, like the air condition we created during this workshop and all DDC Suite templates, are using strong groups and prefixes. During import we renamed immediately Page description/Groups and FBox properties (Name/Ref) – so each imported template has his own data. Nothing else to do after import (at least a quick check in SymbolEditor)

HDLog – each FBox parameterized for historic data will create depending on the FBox property Name all code and symbols for SWebEditor automatically. Due to strict name convention (point before) we are sure that each FBox has a unique name – which is a must when using DDC Suite (and also HEAVAC FBoxes).

At least the reserved memory in HDLog FBox itself must be maybe increased if we have a lot of historic data – but during compile you'll get maybe an error message with information about HDA. Check last error message and add just both memory information in error text, 24 KB reserved and 16 B additional needed = 40 KB and set the definition in HDLog FBox to this size.

Let's check the alarm numbers, especially if all systems are connected to the same alarm list.

When we check the file "DDC_Alarming.csv" and sort by column A and B then we'll see that a lot of alarm numbers are multi defined.

By default the templates are defined in FBox "AlarmHdr" to use alarm no. 11 as first alarm.

Now we have to reorganize the alarm numbers. Lets check the file and find out how many alarms the air condition and a heating circuit defines. (sort by column "C")

Air condition: 16
Heating circuit: 17

	A	B	C	D	E
1	List_1	1	Alarm_1	FuseGuard 230 VAC missing	
2	List_1	2	Alarm_2	FuseGuard 24 VAC missing	
3	List_1	3	Alarm_3	FuseGuard 24 VDC missing	
4	List_1	4	Alarm_4	FuseGuard phase missing	
5	List_1	5	Alarm_5	FuseGuard control voltage	
6	List_1	11	Alarm_11	AC01_Shop_SupplyAir_Temp lim	
7	List_1	11	Alarm_11	HC01_Outdoor_Ttemp limit high	
8	List_1	11	Alarm_11	HC02_Outdoor_Ttemp limit high	
9	List_1	11	Alarm_11	HC03_Outdoor_Ttemp limit high	
10	List_1	11	Alarm_11	HC04_Outdoor_Ttemp limit high	
11	List_1	11	Alarm_11	HC05_Outdoor_Ttemp limit high	
12	List_1	12	Alarm_12	AC01_Shop_SupplyAir_Temp lim	
13	List_1	12	Alarm_12	HC01_Outdoor_Ttemp limit low	
14	List_1	12	Alarm_12	HC02_Outdoor_Ttemp limit low	

DDC Suite 2.0 / PG5 Building Advanced
Using templates – ToDo's after an import

The air condition is the first system in Fupla – so the default start alarm number 11 can stay as it is.

We have to modify it from second system in Fupla, this will be heating circuit HC01. Jump to Page "HC01;General"

Open adjust window of FBox "Alarm Hdr". First system starts with 11 and needs 16 alarms, so 11+16=27. The next free alarm would be 28.

Lets have some spare and start with 31.

Repeat this with all other systems. Let's take the following base alarm numbers:

HC01: 31

HC02: 51

HC03: 71

HC04: 91

HC05: 111

After compile we should have an alarm list without double defined alarm numbers.

List_1	11	Alarm_11	AC01_Shop_SupplyAir_
List_1	12	Alarm_12	AC01_Shop_SupplyAir_
List_1	13	Alarm_13	AC01_Shop_ExhaustAi
List_1	14	Alarm_14	AC01_Shop_ExhaustAi
List_1	15	Alarm_15	AC01_Shop_SupplyAir_
List_1	16	Alarm_16	AC01_Shop_SupplyAir_
List_1	17	Alarm_17	AC01_Shop_SupplyAir_
List_1	18	Alarm_18	AC01_Shop_SupplyAir_
List_1	19	Alarm_19	AC01_Shop_SupplyAir_
List_1	20	Alarm_20	AC01_Shop_SupplyAir_
List_1	21	Alarm_21	AC01_Shop_ExhaustAi
List_1	22	Alarm_22	AC01_Shop_ExhaustAi
List_1	23	Alarm_23	AC01_Shop_ExhaustAi
List_1	24	Alarm_24	AC01_Shop_ExhaustAi
List_1	25	Alarm_25	AC01_Shop_ExhaustAi
List_1	26	Alarm_26	AC01_Shop_ExhaustAi
List_1	31	Alarm_31	HC01_Outdoor_Ttemp li
List_1	32	Alarm_32	HC01_Outdoor_Ttemp li
List_1	33	Alarm_33	HC01_Inflow_Temp_Tole
List_1	34	Alarm_34	HC01_Inflow_Temp_Tole
List_1	35	Alarm_35	HC01_Inflow_Temp limit
List_1	36	Alarm_36	HC01_Inflow_Temp limit
List_1	37	Alarm_37	HC01_Returnflow_Temp
List_1	38	Alarm_38	HC01_Returnflow_Temp
List_1	41	Alarm_41	HC01_Returnflow_Valve
List_1	42	Alarm_42	HC01_Returnflow_Valve
List_1	43	Alarm_43	HC01_Returnflow_Valve
List_1	44	Alarm_44	HC01_Inflow_Pump_Ser
List_1	45	Alarm_45	HC01_Inflow_Pump_SM
List_1	46	Alarm_46	HC01_Inflow_Pump_SM
List_1	47	Alarm_47	HC01_Inflow_Pump_SM
List_1	48	Alarm_48	HC01_Inflow_Pump_SM
List_1	49	Alarm_49	HC01_Inflow_Pump_SM
List_1	51	Alarm_51	HC02_Outdoor_Ttemp li
List_1	52	Alarm_52	HC02_Outdoor_Ttemp li
List_1	53	Alarm_53	HC02_Inflow_Temp_Tole
List_1	54	Alarm_54	HC02_Inflow_Temp_Tole
List_1	55	Alarm_55	HC02_Inflow_Temp limit
List_1	56	Alarm_56	HC02_Inflow_Temp limit

At least we have to update the SWebText.csv file in Sweb application with the Sweb Alarm AddOn tool.

BACnet – each FBox parameterized for BACnet will create depending on the FBox property Name all code and symbols and objects automatically.

Due to strict name convention (point before) we are sure that each FBox has a unique name – which is a must when using DDC Suite (and also HEAVAC FBoxes).

DDC_BACnet.bnt [CPU002] - BACnet Configurator

Project   Edit   Configuration   View   Help

- HC01_Mode SetPointY4 [AV 36]
- HC01_Mode Switch [MV 3]
- HC01_Outdoor_Ttemp Analog [AI 2]
- HC01_Returnflow_Controller Parameter_D [AV 50]
- HC01_Returnflow_Controller Parameter_I [AV 49]
- HC01_Returnflow_Controller Parameter_P [AV 48]
- HC01_Returnflow_Controller Valve_Max [AV 47]
- HC01_Returnflow_Controller Valve_Min [AV 46]
- HC01_Returnflow_Temp Analog [AI 4]
- HC01_Returnflow_Valve Alm_FeedbackHigh [BI 23]
- HC01_Returnflow_Valve Alm_FeedbackLow [BI 24]
- HC01_Returnflow_Valve Alm_Manual [BI 25]
- HC01_Returnflow_Valve Output [AO 0]
- HC02_HeatPeriod_Scheduler State [BV 5]
- HC02_Inflow_Controller Parameter_D [AV 71]
- HC02_Inflow_Controller Parameter_I [AV 70]
- HC02_Inflow_Controller Parameter_P [AV 69]
- HC02_Inflow_Controller Valve_Max [AV 68]
- HC02_Inflow_Controller Valve_Min [AV 67]
- HC02_Inflow_Pump EnCounter [AV 78]
- HC02_Inflow_Pump Feedback [BI 43]
- HC02_Inflow_Pump Output [BO 3]
- HC02_Inflow_Pump Service [BI 44]
- HC02_Inflow_Pump Workinghours [AV 77]

Number of Objects: 289

DDC Suite 2.0 / PG5 Building Advanced
Using templates – ToDo's after an import

At least we have to update the BACnet configuration which is linked to the program – BACnet.bnt. If we added some FBoxes/templates with BACnet configuration we have to start the DDC BACnet AddOn tool.

It will check if maybe some new objects are created from DDC Suite FBoxes and add them to the linked BACnet.bnt configuration file.

After checking the settings for HDLog, alarming and BACnet we have to rebuild the program to be sure that all files are updated before downloading the program.

Predefined templates

DDC Suite 2.0 / PG5 Building Advanced
Using templates – Predefined templates

HeatCircuit_T1_Inflow_Controlled_1Pump

- Outdoor temp. sensor
- Inflow temp. sensor
- Returnflow temp. sensor
- Pump
- Valve

- Set point calculated via outdoor temp.
- Inflow temp. controlled
- Returnflow temp. max. controlled

Heating circuit

DDC Suite 2.0 / PG5 Building Advanced
Using templates – Predefined templates

HeatCircuit_T2_Inflow_Controlled_2Pump

- Outdoor temp. sensor
- Inflow temp. sensor
- Returnflow temp. sensor
- 2 Pump
- Valve

- Set point calculated via outdoor temp.
- Inflow temp. controlled
- Returnflow temp. max. controlled
- Automatic pump sequencing

Heating cirquit

DDC Suite 2.0 / PG5 Building Advanced
Using templates – Predefined templates

HotWater_T1_Primary_Uncontrolled_Circulation

- Tank temp. (top) sensor
- Tank temp. (bottom) sensor
- Inflow temp. sensor
- Max. temp. limiter
- Pump
- Valve

- Inflow temp. controlled
- Disinfection

Circulation:
- Pump

Hot water

DDC Suite 2.0 / PG5 Building Advanced
Using templates – Predefined templates

HotWater_T2_Primary_Controlled_Circulation

- Tank temp. (top) sensor
- Tank temp. (bottom) sensor
- Inflow temp. sensor
- Max. temp. limiter
- Pump
- Flap

- Start/Stop Inflow temp.
- Disinfection

Circulation:
- Pump

DDC Suite 2.0 / PG5 Building Advanced
Using templates – Predefined templates

AirCondition_T1_PressContr_Master_CoolMixAirHeat

- Outdoor temp. sensor
- Supply air temp. sensor
- Exhaust air temp. sensor
- 2 room temp. sensor
- Room set point correction
- Cooler/Mixed Air/Heater
- Supply/Exhaust fan

- Air condition sequence start
- Master-Sequence cascade
- Heater frost protection
- Summer night cooling
- Room frost protection

DDC Suite 2.0 / PG5 Building Advanced
Using templates – Predefined templates

AirCondition_T2_1speed_Master_CoolMixAirHeat

- Outdoor temp. sensor
- Supply air temp. sensor
- Exhaust air temp. sensor
- 2 room temp. sensor
- Room set point correction
- Cooler/Mixed Air/Heater
- Supply/Exhaust fan

- Air condition sequence start
- Master-Sequence cascade
- Heater frost protection
- Summer night cooling
- Room frost protection

DDC Suite - Documentation

DDC Suite FBoxes are object oriented in functionality. Therefore each FBox contains a function, like "Sensor measurement", and all necessary data point.

So it's possible have a clear description with all parameter for a FBox like:

AirCo01_Outdoor_Temp,ref:FuseGuard

Sensor
- Inp   Value
- AlaHigh
- AlaLow

Recording of the measurement reading with calibration, filtering and limit value monitoring. The connected analogue value can be upgraded according to the type.

PCD2.W220 NI1000 DIN : The raw score is supplied by the card (standard/analogue module/PCD2.W220) and the physical value calculated here
PCD2.W340 NI1000 DIN : The raw score is supplied by the card (standard/analogue module/PCD2.W340) and the physical value calculated here
The physical value is already attached at the input contact.
Conversion: An arbitrary value (mostly from an active sensor) is converted by means of a linear equation.

The recorded measurement reading can be calibrated by specifying an adjustment. Subsequently, the measurement reading is filtered. The frequency of the measurement reading can be set, the smoothing factor states the allotment, which is included in the new measurement reading in case of a measured value alteration.

This could be the detailed description from FBox help, but normally this is too detailed or provides too many possibilities how to use the FBox.

The target is to get a documentation file to hand out to the end user or planer with a general description and all useful parameter settings. And this document should be updated on each build to have always an "up to date" description including parameter.

And this document can be used to add some more general information or pictures from Sweb or ViSi.Plus application.

DDC Suite FBoxes are able to do this. Each FBox supports an external file.

- If the file is not present then no documentation is created from this FBox
- if the file is found then the definition for documentation is parsed in this file

So the external file enables the documentation!

There are 2 kind of files

- files which are parsed from the FBox during build – they are not editable
- Source files, editable e.g. with MS Word and saved in HTML format

The first type of files are structured with a strong name convention

- DOC_ - declares that this file contains documentation information
- DDC_ - identifies that this file is used with DDC suite FBoxes
- "Family_" e.g. Alarming - declares the DDC Suite family
- "FBox" e.g. 1Alarm – defines the FBox itself
- .src – file extenstion

Example: DOC_DDC_Alarming_1Alarm.src

But you don't have to know all these file names or create them by yourself.

DDC Suite 2.0 / PG5 Building Advanced
Documentation

The source files are structured with a strong name convention
- DOC_ - declares that this file contains documentation information
- DDC_ - identifies that this file is used with DDC suite FBoxes
- "Family_" e.g. Alarming - declares the DDC Suite family
- "FBox" e.g. 1Alarm_ – defines the FBox itself
- addendum "Main" - general part, must exist
- addendum "HDLog" this part contains the description about historic data (optional)
- addendum "Alarm" this part contains the description about alarm data points (optional)
- addendum "BACnet" this part contains the description BACnet data points (optional)
- .src – file extenstion

So one FBox can have up to 4 source files (DOC_DDC_FamilyFBox_Main.html, DOC_DDC_FamilyFBox_HDLog.html, DOC_DDC_FamilyFBox_Alarm.html, DOC_DDC_FamilyFBox_BACnet.html)

But you don't have to know all these file names or create them by yourself.

And these source files must be converted into the file the FBox is able to parse during build. This will do the **AddOn tool.**

DDC Suite 2.0 / PG5 Building Advanced
Documentation

You'll find already predefined HTML files in a DDC Suite project in folder "FBox_AddOn" and there in subfolder "Documentation).

It's allowed to edit these files e.g. with MS Word to modify the text, format or colour – but also parameter to be displayed – or not.

There are some special definitions:

- @@&name@@ will display later the FBox property name at this position

- @@&ref@@ will display later the FBox property reference at this position

- @adj_SYMBOL@ displays a parameter (addendum .0p or .1p is the definition if value has to displayed without or with decimal sings)

After changing the file you must save it – but in a special format.

Select from menu "File&Save as" and select file type "Website, filtered (*.htm).

This will produce a slim HTML file – if you save it as default HTML file type – it won't work!

After modifying the source files to your belongings we have to combine the (up to) 4 source files into the *.SRC file which is used from the FBox.

These source files must be converted into the file the FBox is able to parse during build. This will do the DDC AddOn tool.

Start the *.ddc Addon tool by double clicking on it.

The DDC-Suite Addon will be started.

First we have to define where the source files – HTML files – are located.

Click on browse button and navigate to your project, subfolder "FBox_AddOns" and there "Documentation".

Then click on button "Generate". This may take some seconds – the tool has to combine the (up to) 4 HTML files per FBox into 1 SRC file and save it in a special format.

If no MessageBox occurs and the success text in the status strip occurs, the created SRC files should appear in the device folder.

If Conversion Error occurred: One of the document files may be write-protected.

Now start a "Rebuild all" in PG5 project manager.

DDC Suite 2.0 / PG5 Building Advanced
Documentation

Open the document with a double click – by default it will be displayed with your default browser.

You'll see that the FBox property name is also displayed and all the basic parameter (grey table).

Additional – depending on parameterized features – you'll also see

- alarm information (orange table)
- historic data definition (green table)
- BACnet configuration (blue table)

Heading amp;name - | SAIA-BURGESS MURTEN | Microsoft®InternetExplorer |

Address: C:\Promos14\proj\DDC_Suite_V2_T5_EN_AirCo\pcd\CPU002\DDC_Documentation.htm

Motor control 1-stage AC01_Shop_SupplyAir_Fan

Control of a 1-stage motor. The control has 3 levels:

- Manual switch. This switch has the positions Off - Auto - On. The position is set on "Auto" one time after the programme download. If there is no manual switch available, this parameter can be used for instance for a higher level virtual operation. If BACnet is activated, this parameter cannot be used otherwise. This parameter then shows <Auto>, if the resulting system enabling of BACnet is identical with the requirement (regardless of whether the system should be Off or On). If a different operating mode is enforced through BACnet, the corresponding status is displayed (e.g. On or Off)

- Building management system switch. Effective only if the manual control switch is on <Auto>. This switch has the positions Off - Auto - On. This parameter is normally used in the virtual operation in a SCADA or WebPanel.

- Automatic requirements. Effective only if the manual control switch and the building management switch are in auto mode. The requirement usually comes from a start or timing function

Operating data / Maintenance

A metering of the operating hours and the switching cycles take place. When it reaches the set maximum value, a warning signal is issued. This can be reset via a central acknowledgement or separately and only for this drive.

Parameter	Value
Building management system switch (1=Auto, 2=Off, 3=On)	1
Start delay	0.0 s
Number of startings till maintenance signal is issued	2000
Number of operating hours till maintenance signal is issued	5000 Std.

Alarm management

The calculated alarm data points can be logged in an internal alarm list. The number and the defined alarm numbers are listed below.

Alarm designation	Number
Maintenance required	5

BACnet objects

DDC Suite 2.0 / PG5 Building Advanced
Documentation

If you like to edit this file we have to save it once from MS IE.

MS IE displays all parts from all FBoxes – MS Word will only display the first FBox part ….

So we have to do a "save as".

Just save it at desktop as "Documentation" with type "Web Page complete", close MS IE and open the saved file with MS Word – or your preferred word processor.

System description

Now you can add some general text like

Aircondition

This air condition is controlled by room temperature sensor an uses cooler, mixed air and preheater ….

If you like to generate such general description with Fupla you have to use the special FBox "Documentation" – located in family "DDC Initialization".

Open Fupla, first page from "COB General" and place FBox on page.

This FBox supports up to 20 user defined documentation files – they are already existing in folder with HTML files and named

DOC_DDC_UserModule(1..20)_Main.htm

User module 1 is already used for a general description about the format. Let's see what happens.

Select "No.1", save and build program.

DDC Suite 2.0 / PG5 Building Advanced
Documentation

Open file DDC_Documentation.htm and you'll see that a new description (user module 1) has been added at this position.

If you use this mechanism in front of each system (e.g. first page of AC01, first page of HC01 … and so on) then you are able to insert a general description for the different systems and you only have to add then some pictures into the document.

I/O Allocation

As you remember we can assign the digital I/O's online in the adjust parameters of the Fbox.

But how can we get an overview which I/O is used by which Fbox?

After a build you can see the I/O allocation in the following file.

DDC_DI_DI_List.csv

You can double click it to view it with MS Excel

ViSi.Plus

Please follow the teachers advice.

Please:
- use the same symbol names
- use the same group names
- place the FBoxes approx. at the same position
- do not work faster or different even if you are a "frequent ViSi.Plus user"

This workshop will show you some basic mechanism, structured workflow and well structured symbol organisation. Don't be afraid.

You don't
- have to learn all FBoxes during this workshop
- have to be familiar with application programming
- must be a super programmer

If you just learn the mechanism and philosophy you'll understand the advantage SI can have with DDC Suite

DDC Suite 2.0 / PG5 Building Advanced
Syntax and remarks of actions during workshop

Click with left mouse button at this position

Double-click with left mouse button at this position

Click with right mouse button at this position

Follow the green arrow to next step

Example

Type in the blue text into the high lighted green text field

Watch this area

Installation of Visi.Plus

Start installer setupvp_1.5.1.18.exe (version number may differ in case of newer versions) and select language.

A password must be typed in to go on.   saia

(must be in small letters)

Click on "Welcome …" dialog at button Next to continue

Select option I accept the terms in the License Agreement
and click on button Next.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

Use option Automatic (predefined) and click on button Next.

ViSi.Plus is starting to install software.

When finished click on button Next.

At least deactivate checkbox Show Readme and click on button Finish.

Installation is completed.

You can access ViSI.Plus software via
- Start
- Programs
- SAIA ViSi+ 1.5

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

Or via file explorer within
C:\ProMosNT.

ProMos is the product name from
developer MST (Müller System
Technik / Belp – near Bern)

This folder contains subfolders:

bin: folder with all executable modules from ViSi.Plus

cfg: some predefined files for special features – we don't use them

hlp: help files and documentation

proj: project folder – herein all ViSi.Plus projects are located

Start up

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

I recommend to access ViSi.Plus modules via file explorer or if you work often with ViSi.Plus create some desktop shortcuts.

Why? ViSi.Plus is a modular software and during engineering it's not necessary to start always all modules – this will cost time during start up.

Therefore we will start only the modules which are necessary to work with.

Start ProjectCfg.exe – this is the ViSi.Plus project manager.

Within the project manager you can
- define the default project to work with
- start a project
- create, copy, rename or delete a project
- manage license and start options
- backup and restore a project
- a quick start for PG5 or PET (data base from ViSi.Plus)

Creating a new project

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

A new ViSi.Plus DDC Suite project could be started within project manager via „New project ". But then the user has to integrate manually all he'd like to use, e.g. HMI, BACnet, graphics, screens, pictures and so on.

We recommend to start a new project via „Restore", that means we are using a predefined DDC Suite template projects where a lot of settings or pages, templates are already prepared.

Therefore those template projects include everything what can be used in a project, e.g. FBox libraries, Fupla template pages, SWeb template objects

Thereby a project backup will be a little bit larger (e.g. >20 MB) but changes in future in this project may have no compatibility problem, e.g. if in meantime a FBox library has totally changed.

Of course – our template project can be updated with 3rd party libraries and backup-ed as new template project.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

So we are starting with „Restore " – by selecting a project template. Depending on template version or location we have to navigate to the correct folder …

First we have to select the folder wherein the templates are available and click on button OK.

From Visi.Plus version 1.5.1.18 the VP_DDC_Suite_XX_YY templates are available the backup folder of PromosNT (C:\PromosNT\backup)

(Get the newest templates from SBC Support HomePage www.sbc-support.ch within Software/PG5/DDC Suite)

Now you'll see in list Available backups all project templates. Select a project and press "OK" and the template project will be used as base for our Workshop project.

(DDC_Suite_V2_T2_ViSiPlus_International_VER_08.10.08 16.27.zip)

After restoring the template project just select it in project list and click on button OK.

The project manager will terminate and nothing else happens. By clicking on button OK we just define the selected project as default project ViSi.Plus will work with when we start manually some modules.

Of course this is not the way the end user has to start ViSi.Plus! Therefore we have a project starter modules which will start all necessary modules in a defined way.

I recommend to use the method I show for engineering – because is more comfortable.

If we have a look into our workshop project (use Windows File Explorer) we'll see that already some folders are included:

- bmp: all BMP or GIF files must be located here
- cfg: data base files, language depending files
- dat: historic data files
- drv: driver settings
- lic: license file
- log: log files
- mac: macros to be used/created in graphic editor
- pcd: PG5 project
- src: screen file, all pages and pop up windows
- www: web server files

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

If we have a look into the folder pcd we'll see that already a complete PG5 project is embedded:

- CPU-Template : a CPU template which should be used for each new CPE we have to add in this project. (When you open PG5 2.0 this template is shown in red colour to indicate that it is read only.)

- FbdToDms : Containing some information if a ViSi.Plus SCADA system should be used

- FBox_AddOns : Containing some files for some language depending definitions and also some additional features

- FBox_Libraries :Change! See next chapter "FBox libraries" in Guideline "DDC Suite Fupla"

- Fupla_Templates : here you'll find some predefined Fupla pages or systems, to be imported into Fupla

- Sweb_Objects : Graphical objects and adjust objects (pages) for SWeb applications designed with SWebEditor.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

Why?

A ViSi.Plus project does not only contain some pictures and graphics – it should also contain the PG5 project itself because ViSi.Plus can get a lot of information out of a PG5 project!

And ViSi.Plus project manager will backup all sub folders within the ViSi.Plus project – so if you backup your project you'll have always a complete backup – and not only the part PG5 or ViSi.Plus. This makes your project consistent.

The PG5 project within a ViSi.Plus project must always be defined as pcd. ViSi.Plus is looking always to a PG5 project pcd within pcd folder. If you rename the PG5 project ViSi.Plus cannot find the PG5 project anymore – this is a hard restriction.

Basic settings

Start module dms.exe from C:\ProMosNT\bin folder. DMS is the shortcut for Data Management System – the core of ViSi.Plus. Without running DMS you're not able top start other modules – they try to connect a DMS in any way (DMS can also run in network on other PC).

ViSi.Plus is protected by a license file (containing the available modules and a serial number) and a USB dongle (also with a serial number). If the serial number from license file and USB dongle don't match you'll get the dialog at right border.

For testing or engineering you do not need an official license – you can work within 4 hours without restrictions. After 4 hours ViSi.Plus will shut down – but can be restarted immediately again for next 4 hours.

Or you can have PG5 2.0 license which has Visi.Plus Engineering Edition license too. In this case Visi.Plus needs to be restarted once per day and all modules are enabled.

DMS is a service and you normally don't have to interact with this module itself. You'll see 2 bullets in task bar notification area:

-Red one is the DMS itself – this is the RAM based data base

-Green one is the PDBS - this is the disk drive based data base

Now we can start pet.exe. PET is a shortcut for
Process Engineering Tool.

When starting PET first time after installation it will ask
for a PLC-driver. Select the entry SDRIVER.PET.

Finally you see PET application.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

ViSi.Plus can manage PG5 projects so therefore we have to define first time after installation which PG5 version we'd like to use.

Therefore select from menu SAIA-PG5, in context menu Options PG5 and there PG5 Version.

Select from drop down list Promos Communication Driver entry PG5 Version 2.0 and finish by clicking on button OK.

Preparation

OK – basic preparations finished.

What we can see in PET:

- symbol bar with icons with PG5 connectivity

- tab Process objects where all objects from data base are listed

As we see there are no process objects – we have to create some. But this we'll do in a normal way – means engineering a Fupla and getting the data from the Fupla into ViSi.Plus data base.

Therefore just click on symbol PG5

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

PET will start the PG5 project manager.

We have to select the desired project PCD within the ViSi.Plus project in Project/ Open

Back to Guideline "DDC Suite Using Templates":

Back to PET.

Importing data from Fupla

Be aware that at least the Fupla has been compiled – I recommend to build the program successfully. During compile or build a file is created which is necessary for ViSi.Plus import functionality!

Select from menu SAIA-PG5 entry Import Labels.

In next dialog you see that ViSi.Plus also knows all CPU's in PG5 project. Select CPU002 from list and click on button Selection for PG5 Symbols.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

A progress dialog is shown and ViSi.Plus is parsing a file in PG5 project in CPU002 to find all DDC Suite FBoxes and the depending group structure.

Maybe after import you don't see any process objects – in this case we have to set a filter. Select Options from menu and then Filter.

Select in dialog from drop down list PLC-Name entry CPU002 and click on button OK.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

Now we see that PET imported 81 process objects.

If you'd like – count the DDC Suite FBoxes (with internal data) in Fupla – and you'll count 81 FBoxes.

In fact – PET handles DDC Suite FBoxes as object – not as single data points.

PET - Process Engineering Tool (SAIA ViSi+) 1.5

File   Edit   Templates   Options   SAIA-PG5   View   About

	Description	DMS-name	Object type	Scheme
57	HC04 System Status	CPU002:HC04:System:Status	vloAmpel	
58	HC04 System HeatPeriod	CPU002:HC04:System:HeatPeriod	DdcSpGener21	
59	HC04 System Mode	CPU002:HC04:System:Mode	DdcEnHeatc21	
60	HC04 Outdoor Temperature Sensor	CPU002:HC04:Outdoor:Temperature:Sensor	DdcAnGener21	
61	HC04 Inflow Temperature Sensor	CPU002:HC04:Inflow:Temperature:Sensor	DdcAnGener21	
62	HC04 Inflow Temperature Tolerance	CPU002:HC04:Inflow:Temperature:Tolerance	DdcAlGener27	
63	HC04 Inflow Temperature Controller	CPU002:HC04:Inflow:Temperature:Controller	DdcReContr21	
64	HC04 Inflow Pump Alarming	CPU002:HC04:Inflow:Pump:Alarming	DdcAlMotor21	
65	HC04 Inflow Pump Control	CPU002:HC04:Inflow:Pump:Control	DdcCoPump21	
66	HC04 Returnflow Temperature Sensor	CPU002:HC04:Returnflow:Temperature:Sensor	DdcAnGener21	
67	HC04 Returnflow Temperature Controller	CPU002:HC04:Returnflow:Temperature:Controller	DdcReLimit21	
68	HC04 Returnflow Valve Control	CPU002:HC04:Returnflow:Valve:Control	DdcCoValvd21	
69	HC05 System Manual	CPU002:HC05:System:Manual	DdcGeManua21	
70	HC05 System Status	CPU002:HC05:System:Status	vloAmpel	
71	HC05 System HeatPeriod	CPU002:HC05:System:HeatPeriod	DdcSpGener21	
72	HC05 System Mode	CPU002:HC05:System:Mode	DdcEnHeatc21	
73	HC05 Outdoor Temperature Sensor	CPU002:HC05:Outdoor:Temperature:Sensor	DdcAnGener21	
74	HC05 Inflow Temperature Sensor	CPU002:HC05:Inflow:Temperature:Sensor	DdcAnGener21	
75	HC05 Inflow Temperature Tolerance	CPU002:HC05:Inflow:Temperature:Tolerance	DdcAlGener27	
76	HC05 Inflow Temperature Controller	CPU002:HC05:Inflow:Temperature:Controller	DdcReContr21	
77	HC05 Inflow Pump Alarming	CPU002:HC05:Inflow:Pump:Alarming	DdcAlMotor21	
78	HC05 Inflow Pump Control	CPU002:HC05:Inflow:Pump:Control	DdcCoPump21	
79	HC05 Returnflow Temperature Sensor	CPU002:HC05:Returnflow:Temperature:Sensor	DdcAnGener21	
80	HC05 Returnflow Temperature Controller	CPU002:HC05:Returnflow:Temperature:Controller	DdcReLimit21	
81	HC05 Returnflow Valve Control	CPU002:HC05:Returnflow:Valve:Control	DdcCoValvd21	

Process-objects   Detail view   Digital Signals   Analog Signa

Press F1 to get help   DdcEnHeatc21   Filter: Templates   07:25:53

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

1. Each FBox has corresponding symbols in a group structure. Here it's PCD.Monitoring.

2. PET identifies during import FBox InitLIB and thru import rules PET creates in data base an object of type DdcInLibra21.

3. PET also identifies the corresponding group structure – and this is used to create a unique data base name – the DMS-Name. PET adds the CPU name as prefix because this group structure maybe also exists in another CPU in this project. The DMS-Name must be unique and is write protect.

4. The Description is a text shown in tool tip or pop up windows to identify the object the end user is operating. The Description can be modified.

PCD_Monitoring

Init LIB
AckAla ResAla
AckMt ResMt
WD Err
OutTemp Bat
Pulse

Symbols

Group/Symbol	Type
PCD	GROUP
iAlaFuseGuard	F
iEverythingOK	F
Monitoring	GROUP
SmQuit	F
WartQuit	F
HistQuit	F
HistFehler	F

	Description	DMS-name	Object type
1	PCD Monitoring	CPU002:PCD:Monitoring	DdcInLibra21
2	PCD AntiblockProtection	CPU002:PCD:AntiblockProtection	DdcInAntib21
3	PCD FuseGuard	CPU002:PCD:FuseGuard	DdcAlGener20

Click at tab Detail view to get more details of the objects (you also can say FBoxes).

Scroll down to line 70. This should be the almost top line.

As you can see this is the FBox with group structure

AC01 ExhaustAir Temperature Sensor

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

You see that all symbols from a FBox are also in PET available.

And thru the import rules they get in PET automatically a default trending or alarming definition.

That means by importing you get automatically additional predefined options in ViSi.Plus.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

To make it clear:

- within up to 98% of all cases you don't have to parameterize manually data points for trending or alarming in ViSi.Plus with DDC Suite FBoxes

- if you use always dedicated FBoxes for alarming or trending – even ig they produce a data overhead in Fupla – you never have to parameterize those functionalities

- Of course you have to modify the default alarm texts or priorities – but this can be done "on the fly" during runtime of ViSi.Plus

So this import feature within ViSi.Plus saves you a lot of work!

Addressing the resources

ViSi.Plus is getting data from a PCD via S-Bus communication. Therefore each data point must have a unique address like R 123 or F 234.

By default the symbols in Symbol Editor have no address so we have to define the address.

Sensor	GROUP	
MessTyp	R	(5) Selection of the c
RohwertX2	R	(5) Maximum raw valu
FilterFaktor	R	(5) Factor for influenc
RohwertX1	R	(5) Minimum raw valu
Istwert	R	(1) Physical value of t
GwUnten	R	(4) Low limit, for pass
IstwertY1	R	(5) Minimum physical
GwOben	R	(4) High limit, for pass
FilterZeit	R	(5) Scanning time of t
SmGwOben	F	(1) Alarm Value bigge
Korrektur	R	(5) Correction value i
SpgGrp	R	(5) Associated voltag
IstwertY2	R	(5) Maximum physical
SmGwUnten	F	(1) Alarm Value less t

Let's do this in PET – because here it takes 2 minutes to sort and address all data points. No matter if you address 100 or 10.000 data points.

Activate tab Digital Signals. Here you'll find all flags to be communicated with PCD. You also see that PET already took over the Channel settings and S-Bus station number.

But the addresses are still 0 for each data point - because in Symbol Editor nothing has been defined.

Within this workshop we do not have a lot of data points and USB communication is quite fast – that means that we can get all information within 1 second into ViSi.Plus if we are online.

But if this will be a big application with 10 PCD and each PCD with up to 5 air conditions than you can imagine that it may tale probably some second if an alarm is detected by ViSi.Plus. Therefore we should organise the data points into groups (telegrams) to optimize the communication load.

This is already prepared, just double click on column header "Comment".

You see that the comment contains a number from 1 to 5 in brackets. This number indicates the communication philosophy, e.g.

(1) communicate as fast and often as possible (alarm/view values) to have "real time" behaviour

(2) once per 5 second, e.g. feedback information having by default some delay

(3) once per minute (or less), e.g. working hours or clock status

(4) only at start up and then never, e.g. SetPoints (update them if HMI or SWeb in use)

(5) only at start up and then never, always commissioning data points (open/closed and so on)

	Comment	Channel	Station	T-Nr.	Typ	Address	
mp:Al	(1) Störmeldung Motors	S-Bus_USB	2	11	Flag	0	N
n:Alar	(1) Störmeldung Motors	S-Bus_USB	2	11	Flag	0	N
Alarm	(1) Störmeldung Motors	S-Bus_USB	2	11	Flag	0	N
:Alarm	(1) Störmeldung Rep.-S	S-Bus_USB	2	11	Flag	0	N
an:Ala	(1) Störmeldung Rep.-S	S-Bus_USB	2	11	Flag	0	N
mp:Al	(1) Störmeldung Rep.-S	S-Bus_USB	2	11	Flag	0	N
n:Alar	(1) Störmeldung Rep.-S	S-Bus_USB	2	11	Flag	0	N
Alarm	(1) Störmeldung Rep.-S	S-Bus_USB	2	11	Flag	0	N
an:Co	(2) Anzeige Ansteuerun	S-Bus_USB	2	11	Flag	0	N
n:Cont	(2) Anzeige Ansteuerun	S-Bus_USB	2	11	Flag	0	N
:Contr	(2) Anzeige Ansteuerun	S-Bus_USB	2	11	Flag	0	N
mp:Co	(2) Anzeige Ansteuerun	S-Bus_USB	2	11	Flag	0	N

Now let's define the real address. Scroll up to first line and type in 100 in Column Address/Row 1

	T-Nr.	Type		Logic
1	11	Flag	1 00	Normal
2	11	Flag	0	Normal
3	11	Flag	0	Normal
4	11	Flag	0	Normal
5	11	Flag	0	Normal
6	11	Flag	0	Normal
7	11	Flag	0	Normal

Click on column header Address – the whole column is getting selected.

Type	Address	Logic
Flag	100	Normal
Flag	0	Normal
Flag	0	Normal
Flag	0	Normal
Flag	0	Normal
Flag	0	Normal
Flag	0	Normal
Flag	0	Normal
Flag	0	Normal

Click with right mouse button on column header Address – all flags getting ascending numbered.

That's quite easy ☺

Type	Address	Logic
Flag		Normal
Flag	101	Normal
Flag	102	Normal
Flag	103	Normal
Flag	104	Normal
Flag	105	Normal
Flag	106	Normal
Flag	107	Normal
Flag	108	Normal

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

We have to repeat the same procedure at tab Analog Signals. Herein all registers are listed. Activate tab.

Double click at column header Comment to sort by communication priority

Now let's define the real address. Scroll up to first line and type in 200 in Column Address/Row 1

Click on column header Address – the whole column is getting selected.

Click with right mouse button on column header Address – all registers getting ascending numbered.

Finally save your work by clicking at disk symbol.

Now we must get the address information back into PG5 Symbol Editor. Select from menu SAIA-PG5 entry Export Options.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

Within this big dialog we're only interested in settings Filter. We set the filter already to CPU002 – so it should be already activated.

If not, select from drop down list entry CPU002. The text fields DMS-Filter and Export-Filter (beginning of DMS-Name) should be set automatically to CPU002 – and also the checkbox Delete character at the beginning should be activated.

Check this and click on button OK.

Now we are sure that only data points from this CPU will be exported – otherwise e.g. you'll have no or too much information in export file – and this leads into data chaos in Symbol Editor ….

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

Select from menu SAIA-PG5 entry Resourcenliste
für FUPLA generieren (.rxp)

By default you should be located in project base
folder.

open folder pcd

open folder CPU002 and click
on button Save. An export file
pet.rxp will be created.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

You may see a progress dialog during exporting data into file

And finally the information that the file has been created. Close dialog.

That's all what we have to within PET. Close PET.

At least we have to import the file from PET into PG5 Symbol Editor.

Switch to Fupla / Symbol Editor.

Click with right mouse button into symbol editor, select Advanced and in sub menu entry Import Symbols…

Within dialog Import Symbols select from drop down list Files of type entry ViSiPlus Symbol Files (*.rxp)

Select file pet.rxp and start import by clicking on button Import.

You'll see a dialog to inform you that there is NO UNDO possible.

I do Not see this pop up in PG5 2.0 why !!!!

Fupla detects during import that there is already a symbol existing in symbol table with name XYZ and the same symbol is also existing in file to be imported.

For sure – because we imported this information from Fupla into PET and now back – but we gave the data points an address.

Therefore select option Replace the symbol address

and activate checkbox Use selected action for all multi-defined symbol name

Otherwise you have to repeat this dialog hundreds or thousand times …

The import may take some seconds but finally you should see all groups with FBox data addressed:

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

The import may take some seconds but finally you should see all groups with FBox data addressed:

You'll see also some groups with symbols without address – they are only internal symbols used in connectors.

At least all symbols in almost all groups should be addressed.

Save, build program and download into PCD and go online with Fupla.

PG5 Building Advanced / DDC Suite 2.0
DDC Suite and ViSi.Plus

Go online

Title of this presentation

49

Till now we only imported FBoxes into ViSi.Plus (via PET) – addressing data points and exporting back into Fupla.

1st – lets start the ViSi.Plus communications driver. Start SDriver.exe from folder C:\ProMosNT\bin

SDriver is a service and also available in task bar notification area.

Now we have to define the physical communication layer. Therefore we have to access the SDriver window. Click with right mouse button on SDriver symbol in task bar and select Show.

SDriver list on left side all Channels defined within PET. Of course – we got this information during import from PG5 but SDriver handles this only as name. The real physical layer must be defined now.

Good to know:

⬛ Communication disabled

⬛ trying to initiate communication

⬛ communication established

⬛ communication failed

Click with right mouse button on channel, dialog
For detailed communication settings will prompted

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

Click at button Setup SCOMM to define the physical layer.

Select in drop down list Channel / Name entry S-Bus USB

Deactivate checkbox PGU

And set S-Bus Station to 2

Finish by clicking on button OK.

At least activate checkbox Driver activated, set parameter
Poll interval (between telegram read accessess) ms: 200
and finish by clicking on button OK.

After a view seconds you should see a green PCD2 symbol in channel list – you are connected with PCD

Close SDriver window. SDriver will work in background. From this moment we can get already a lot of information out of the PCD without additional work.

Alarm management

Let's check if we have some active alarms.

1st – lets start the ViSi.Plus Alarm Manager. Start AlmMng.exe from folder C:\ProMosNT\bin. AlmMng is a service and also available in task bar notification area.

2nd - to view the alarms we have to start the Alarm viewer. Start AlmView.exe from folder C:\ProMosNT\bin

The alarm viewer is the front end for End user to view/acknowledge alarms and/or search in alarm history

SAIA ViSi+ AlarmViewer - 1.5

File   Edit   Filter Settings   View   Help

Q   P1 P2 P3 P4 P5 All   X   ?

8 Alarms not acknowledged of 8 alarms | Status

1	09.10.2008 15:01:02 Kommt CPU002:AC01:SupplyAir:Temperature:Sensor:SmGwUnten Grenzwert unterschritten	come
2	09.10.2008 15:01:02 Kommt CPU002:AC01:ExhaustAir:Temperature:Sensor:SmGwUnten Grenzwert unterschritten	come
3	09.10.2008 15:01:02 Kommt CPU002:AC01:Room:Temperature_1:Sensor:SmGwUnten Grenzwert unterschritten	come
4	09.10.2008 15:01:02 Kommt CPU002:AC01:Room:Temperature_2:Sensor:SmGwOben Grenzwert überschritten	come
5	09.10.2008 15:01:02 Kommt CPU002:AC01:Preheater:Frost:Protection:SmRlTemp Frostwarnung Rücklauftemperatur	come
6	09.10.2008 15:01:02 Kommt CPU002:AC01:Preheater:Frost:Protection:SmFrost Frostgefahr Heizregister	come
7	09.10.2008 15:01:02 Kommt CPU002:HC01:Outdoor:Temperature:Sensor:SmGwOben Grenzwert überschritten	come
8	09.10.2008 15:01:02 Kommt CPU002:HC01:Returnflow:Temperature:Sensor:SmGwOben Grenzwert überschritten	come

Current alarms \ Alarms

Press F1 to get help | Connection : Local | 09.10.2008 | 15:07:17

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

First click on column header and than on button Q to acknowledge all alarms.

All alarms should be acknowledged.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

Switch to Fupla and jump to page HC01;
Controlling/Pump.

Page Navigator

- COB HeatCircuit
  - Page 1 : HC01; General
  - Page 2 : HC01; Alarming
  - Page 3 : HC01; Operating mode
  - **Page 4 : HC01; Controlling/Pump**

Open adjust window from FBox AlaMotor1
and modify in group [--- Motor protection --
-] parameter Normal input state from
opened to closed and write into PCD.

HC01_Inflow_PumpAla.ref:FuseGuard
AlaMotor1

HC01.Inflow.Pump.iFeedbackRun

Adjust: Motor 1 speed 2.0

| Read All | Write all | Set Defaults | Info | Help | Save | Close |

[--- Motor protection ---]				
Digital input	>	-1	< >	-1	0π
Acknowledgement mandatory	>	No	< >	No	0π
Normal input state	>	opened	< >	opened	0π
Alarm suppression	>	for appl. vltg.	< >	for appl. vltg.	0π
Alarm status				ok	0π

You will see immediately in alarm viewer a
new entry.

6	09.10.2008 15:11:39 Quit CPU002:AC01:Preheater:Frost:Protection:SmFrost Frostgefahr Heizregister		quit
7	09.10.2008 15:11:39 Quit CPU002:HC01:Outdoor:Temperature:Sensor:SmGwOben Grenzwert überschritten		quit
8	09.10.2008 15:11:39 Quit CPU002:HC01:Returnflow:Temperature:Sensor:SmGwOben Grenzwert überschritten		quit
9	09.10.2008 15:43:05 Kommt CPU002:HC01:Inflow:Pump:Alarming:MotSm Motorschutz ausgelöst		come

◄ ► \ **Current alarms** ⅄ Alarms /

Press F1 to get help   Connection : Local   09.10.2008 15:43:20

8	09.10.2008 15:11:39 Quit CPU002:HC01:Returnflow:Temperature:Sensor:SmGwOben Grenzwert überschritten		quit
9	09.10.2008 15:45:01 Geht CPU002:HC01:Inflow:Pump:Alarming:MotSm Motorschutz ausgelöst		go

◄ ► \ **Current alarms** ⅄ Alarms /

Historical data

Let's check if we have some active alarms.

1st – lets start the ViSi.Plus Historic Data Manager. Start HDAMng.exe from folder C:\ProMosNT\bin. HDAMng is a service and also available in task bar notification area.

The historic data will be automatically recorded – even if we don't see anything. The data can be viewed later on within the ViSi.Plus screens or with module pCHart. Until we have enough data to view let's check if data will be recorded.

Therefore we have to access the HDAMng window. Click with right mouse button on HDAMng symbol in task bar and select Show.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

First moment the window stays empty – we have to activate the debug mode by clicking on in symbol bar on "Bug" button.

At least after 60 seconds you'll see a lot of entries recorded. By default it defined to record values after 60 seconds – additional on difference of 1.0 (unit)

Deactivate debug mode by clicking again in symbol bar on "Bug" button.

Close HDAMng window. HDAMng will work in background.

User Front-End

The systems must be visualized – therefore have to use the graphics editor - GE.

1st – lets start the ViSi.Plus graphics editor. Start GE.exe from folder C:\ProMosNT\bin.

The graphics editor will start up in runtime mode and you'll see already a predefined start page.

We have predefined a whole navigation structure to start immediately with drawing the systems instead of thinking about "how should I do the navigation, which data must be displayed, where can I find the information if a new alarm raised .."

DDC Suite 2.0 / ViSi.Plus Template
DDC Suite and ViSi.Plus

The screen is separated in 4 areas:

Title bar – use for headline

Menu structure

Painting – use for schematic drawing (process image)

Status bar

Status bar displays basic information:

- Alarm bell icon, grey if there is no alarm, red if min. 1 alarm active (click for alarm list view)

- number of new (not yet acknowledged) alarms and total active alarms

- last alarm text

- Icon "settings" > click to setup menu structure

- Icon "talk" > click to setup user (both icons only visible if user right level 16 active)

- User name logged on
- "User" Icon > click for login dialog
- Or press CTRL+L to get the login window
- Username: Admin
- Password: admin

SAIA ViSi+ - Login 1.5
Username
Password
Licensed from **Demolizenz Visi.Plus**
Saia Burgess Controls AG
©2009 Saia-Burgess Controls AG
Login
Cancel
Licence
Logout

- Date and time

11   New alarms   7   Total alarms
04.10.2010 15:04:04 disparue Alarme AI_Tempo_0 Text alarme par défaut

Admin

25.11.10
07:53:02

DDC Suite 2.0 / ViSi.Plus Template
DDC Suite and ViSi.Plus

Menu structure supports up to 10 main categories. Clicking on category opens page menu. Each category support up to 15 pages, so 150 pages are already available.

Adapting menu structure can be done in runtime mode by clicking on "settings" icon in status bar

Select number of visible main categories. Changes will update view immediately.

Category name and pages of category can be modified by clicking on a category

Select number of visible pages of category. Text of category and pages can be modified by clicking on texts.

An icon indicating the basic meaning (air condition, chillers, electrics …) of category can be defined by clicking on icon in icon bar.

View is updated immediately

DDC Suite 2.0 / ViSi.Plus Template
DDC Suite and ViSi.Plus

Menu structure can be used in 2 different modes:

Permanent visible

- reduces area for painting
- but navigation is easy, always visible

Pop up menu

- enlarges area for painting
- but navigation always via pop up window

Style can be modified in edit mode. Each page is using a "Master" (= background page). 2 predefined background pages are available:

Background_with_NavBar.psc
Background_with_PopUp_NavBar.psc

Go to "Edit mode" with pressing "E" and click on the window and press the property button to open the Window-properties popup. Select in property "Master" the style for the pages (page by page) if you like to have not always a unique menu structure style.

If you like to change all pages from "permanent" to "pop up" menu structure style open file "Background_with_PopUp_NavBar.psc"

And save file (overwrite) as "Background.psc"

Pages are defined to use "Background.psc" as master, so all pages are updated immediately.

Only "Home.psc" (main page) is defined to use "Background_with_NavBar.psc" to display menu structure by default, but this can also be modified.

Screen resolution for pages is predefined for 1280x1024 pixel, so if a different resolution is needed just adapt the background pages (e.g. enlarge titel/status bar or/and move status bar up/down)

Drawing picures

DDC Suite Fboxes offer in ViSi.Plus predefined graphical objects and related adjust/pop up windows. In Edit mode you'll see on left side a catalogue bar. The default view (icons) is not always helpful, just change view by clicking with right mouse button on empty space in catalogue bar an select from context menu "View/Names Only".

DDC Suite 2.0 / ViSi.Plus Template
DDC Suite and ViSi.Plus

Catalogue bar contains also 8 families (same as FBox selector) and offers for (nearly) each FBox an object with same name of FBox selector. Sometimes you'll find 2 or more entries for a FBox, e.g. "System Heating circuit" and "System Heating circuit set point".

In this case the object with "short" name is the default object, displaying the main information (System On/Off ..), the "extended" ("set point") object is just an adaption and displays in this case the set point – but both objects will show the same adjust window when clicking on it.

DDC Suite 2.0 / ViSi.Plus Template
DDC Suite and ViSi.Plus

You just have to select the corresponding object for an FBox and initialize it with the related object from ViSi.Plus data base. All adjust windows are predefined, clicking on graphic object will open a pop up.

Let's check the first Fupla page (General) of our air condition. There are 3 FBoxes to be displayed in ViSi.Plus:

- Alarm header – no Visi.Plus template
- Manual operation
- Sensor - Outdoor air temperature

Lets find out how the FBox is named in FBox selector. Click on FBox "ManualOP" and in FBox selector we see that "Manual Operation 2.0" is selected from family "DDC General"

Lets find a related ViSi.Plus object in GE. Select catalogue "DDC General" and check if there is an object "Manual operation" available.

Let us select the Air Condition → AC01 page in online and press "E" to go to edit mode.

Drag and drop the object into the page. When you drop the object a dialogue will be shown. Remember – during import data from Fupla ViSi.Plus treated each FBox as object. And now ViSi.Plus detects that you would like to display (some) data from FBox "Manual operation" …

ViSi.Plus is checking its data base (which is in fact the whole Fupla data) if there is a "FBox" Manual Operation 2.0 available. A least ViSi.Plus found one for Air Condition 01 (AC01) and three for Heat Circuits – listed in field "to"

Now we just have to assign the AC01:System:Manual FBox to this object. Click on the entry and then OK.

Move the object into the upper left area. If you click away from the object it disappears (Its visibility property is off). To show at least the outline of the object please go to the "View menu" and check the "Show Invisible Objects".

Save (short key "CTRL-S") and switch to runtime mode by pressing key "E".

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

When you move the mouse over the object and wait 2 seconds you'll see a tool tip – this tool tip displays the object name – and the object name is generated during import by using the group structure of the FBox symbols.

Click at the object and a pop up window appears. This is like the "Adjust Window" from Fupla – you don't have to engineer any data point to be viewed – just select an object from catalogue bar and connect it with the related FBox from you PG5 project.

All other detailed windows are predefined and managed by ViSi.Plus.

Also the pop up window displays the object name = symbol group definition!

Play around to see that the Manual Operation is really connected with the Fupla FBox

On/Off drives/motors

AC01 System Manual

On/Off drives/motors

DdcGeManua21

AC01 System Manual

On/Off drives/motors

Manual operation 1 Reset

Continous drives

Manual operation 0 Reset

Controller

Manual operation 0 Reset

Close pop up window and switch back to edit mode by pressing key "E". Now its easy to engineer the other FBoxes from first page.

Now we get a list with a lot of entries. Of course all "Sensor" FBoxes in ViSi.Plus database will be displayed.

Drag&Drop object "Sensor" from "ddc analogue values 2.0" into page and connect it with FBox.

We can select it from the list, but maybe you'll select a sensor from a wrong system.

So we can set a filter in the dialog (bottom). The filter is case sensitive.

Type in CPU002:A

And you'll see only the FBoxes from the Air Condition 01.

Connect it with FBox "AC01:Outdoor:Temperature:Sensor"

Result:

So this Fupla page is done – next page contains 2 FBoxes:

- Or – no Visi.Plus template
- System status

Select from catalogue bar "DDC system and clocks" the object "System status".

So this Fupla page is done – next page does not contain FBox which has template. So let us go to the page "Filter". There are 3 FBoxes there:

▪ 3 times AlaDelay – Delayed Alarm

Drag and drop from catalogue bar "DDC alarming" the object "Delayed Alarm filter right" 2 times and "Delayed Alarm filter left" once.

The "right" ones should be connected to the Outdoor filter and the Supply air filter while the "left" one should be connected to the Exhaust air filter.

So this Fupla page is done – next page contains 4 Fboxes should be placed and connected as we have done before:

- System 1sp
- Fan Start
- Fan special
- Selection

Drag and drop from catalogue bar "DDC system and clocks" the objects "System 1 speed", "Fan start" and "Fan Special func.".

Drag and drop from catalogue bar "DDC analogue values" the object "Selection".

And connect them to the FBoxes.

Go through all the Fupla pages of the Air Condition 01 and place a the corresponding graphical element for the Fboxes onto the screen.

Connect the graphical elements to the Fboxes.

Now the whole air condition is done. Switch to runtime mode an play a little bit with the objects = FBoxes to see which possibilities you have and what's already predefined within ViSi.Plus e.g. historic data or alarming.

At the end some static drawings

This should be placed on to the screen and should be organised nicely.

If something is missing it is easy to place and connect afterwards.

Engineering with templates

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

The air condition was programmed in Fupla to our own belongings and therefore we had to engineer the drawing in ViSi.Plus step by step.

But we also used in Fupla a template – heating circuit – 3 times. Navigate to "Home" – "Heating Circuit" and finally "HC01" and switch to edit mode.

As you see we are exactly on the page where we would like to draw the process image of Heat Circuit 01.

In Fupla we used a template – and this template was predefined by DDC Suite. You'll also find a related template in ViSi.Plus to such Fupla templates.

Click on icon "Load picture" (or menu "File/Open"). The Fupla template was named "HeatCircuit_Inflow_Controlled_1Pump".

Select the file with the same name and click on the Open button.

When the file is opened you can see the predefined heating circuit template. It's created in the same way as we did before – only the connection to FBoxes is not done.

Press short key "CTRL+A" to mark all objects

Press short key "CTRL+C" to copy them to clip board

Press short key "CTRL+F4" to close the picture

Press short key "CTRL+V" to paste it into our structure – and move the whole template that it fits into the page

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

The object on the left side are not necessary if we use permanent navigation bar .

Please select the unnecessary objects and delete them.

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

Now we have to assign to each object an FBox from out heating circuit HC01.

Unselect by clicking into an empty place on page and click on first object "Requirement" to mark it.

Press "Space bar" (key "space") and the dialogue to assign a FBox will be displayed.

The filter "CPU002:A" is still active – we have to modify it. Change filter into "CPU002:HC01" to be sure that we select always objects from HC01.

Assign the FBox to object and click OK

Heat circuit

Change Initialisation Template-Objects

Change initialisation from

DdcAnGener21

to

CPU002:HC01:Inflow:Temperature:Sensor [HC01 Inflow Temperature Sensor]
CPU002:HC01:Outdoor:Temperature:Sensor [HC01 Outdoor Temperature Sensor]
CPU002:HC01:Returnflow:Temperature:Sensor [HC01 Returnflow Temperature Sensor]

OK

Cancel

Filter CPU002:HC01

New BMO

Repeat this for objects you see. If you don't know if a object can be assigned to a FBox – just click on it, press key "space".

If the first entry "Change initialisation from" is empty then this object is a simple static object – nothing to do.

You see that it's easy to use also templates in ViSi.Plus. It takes maybe 1 minute to assign all objects to related FBoxes.

But we have to do still 2 more heating circuits. HC02 to HC03.

Here we can use a mechanism which is quite close to the Fupla import mechanism.

Save the page (CTRL+S)

Change Initialisation Template-Objects

Change initialisation from

to

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

Be sure that all objects are connected to a FBox from HC01.

Mark all (don't use CTRL+A) objects, use key "shift" to add objects to selection.

Be sure that you have marked all objects and graphics from HC01.

Switch to runtime mode, navigate to HC02 and switch back to edit mode.

Click on icon "Load macro" (or menu File/Load macro)

Select the macro we saved before:
heat_circuit.pmc and press "Open"

DDC Suite 2.0 / PG5 Building Advanced
DDC Suite and ViSi.Plus

ViSi.Plus will import the macro – but it detects also that all objects in this macro are assigned to a unique group structure

CPU002:HC01

So here we have the possibility to change all assigned objects in one step from HC01 into HC02.

Click on button "**>**"

(you can also modify the ending HC01 into HC02 immediately in field "Reinitialisiation" – but this may give the chance of type writing errors)

The data base structure navigator opens and now we can navigate to the "system" HC02 to assign all the objects.

A single click on entry "CPU002" – the next layer (group) will be listed.

A single click on entry "HC02" and then button OK.

Check if field "Reinitialization" is changed into

CPU002:HC02

And press OK.

That's all. Check if the objects are assigned to HC02 – do this in runtime mode.

Reinitalisation Macros

Initialisations

CPU002:HC01:Inflow:Pump:Alarming:Bildwechsel
CPU002:HC01:Inflow:Pump:Alarming:Bildwechsel
CPU002:HC01:Inflow:Pump:Alarming:Bildwechsel
CPU002:HC01:Inflow:Pump:Alarming:Bildwechsel
CPU002:HC01:Inflow:Pump:Alarming:Bildwechsel
CPU002:HC01:Inflow:Pump:Alarming:Bildwechsel
CPU002:HC01:Inflow:Pump:Alarming:FbAla
CPU002:HC01:Inflow:Pump:Alarming:ManAla
CPU002:HC01:Inflow:Pump:Alarming:MotAla
CPU002:HC01:Inflow:Pump:Alarming:PfbAla
CPU002:HC01:Inflow:Pump:Alarming:SsAla
CPU002:HC01:Inflow:Pump:Control:Bildwechsel

OK

Cancel

Greatest correspondence

CPU002:HC01

Reinitialization

CPU002:HC02

Now you can imagine how many time you save when you have to draw big air condition with up to 50, 60 or 100 objects.

Any problems?

Yes: rewind to first slide and repeat all lessons …

Everything OK: Congratulation

DDC Suite - advanced – detailed information

Stephan Hintze / 05.12.2008

Overview

DDC Suite FBoxes are able to implement additional external source files. Those files are available for SWebAlarming, BACnet, Documentation and ExtraAddOn.

Depending of FBox functionalities it can be any combination of those files. In the following description you'll have on top the FBox name and then a definition which files are supported and their file names:

BACnet: BAC_DDC_InitLibrary.src
Doc-File: DOC_DDC_InitialisationLibrary.src
AddOn: AddOn_DDC_InitialisationLibrary.src

These files are normally predefined and used to change some texts (alarming) or optimize BACnet object – but they can also be used to implement additional functionality into the FBox or to improve functions. This can be handled like an object inherit mechanism

In fact if you improve e.g. for FBox "1 Alarm" the functionality in an external file – all FBoxes in the program will inherit this new improvement. This is a very powerful possibility.

NOTE: There are only AddOn files for the FBoxes that have online I/O addressing.

DDC Suite Advanced – Overview

For improving functionality you need the internal symbolic definitions for the In/Out of FBox parameter and also the internal data available in adjust window. This information is the content of the following pages. You'll find a page for each FBox and the description

-FBox IN label an the internal symbol

-FBox itself

-FBox OUT label an the internal symbol

-FBox internal symbol and the default symbol name which is used in SymbolEditor

Always just use the names in column "Symbol" within the extra files!

Label	Symbol
AckAla	in_QuitStoerung
AckMt	in_QuitWartung
WD	in_WatchDog
OutTemp	in_Aussentemp

Init LIB

AckAla ResAla
AckMt ResMt
WD Err
OutTemp Bat
Pulse

Label	Symbol
ResAla	out_ResetStoerung
ResMt	out_ResetWartung
Err	out_Error
Bat	out_Batterie
Pulse	out_SekundenPuls

Symbol	Default Symbol
stc_QuitStoerung	SmQuit
stc_QuitWartung	WartQuit
stc_QuitDiagnose	HistQuit
stc_DiagnoseSm	HistFehler
stc_ResetStoerungPuls	SmQuitHwImp
stc_DiagXob	HistMeldung
stc_DiagPrgL	HistProgLine
stc_DiagIdx	HistIndex
stc_DiagCobL	HistCobLine
stc_DiagNL1L	HistCall1
stc_DiagNL2L	HistCall2
stc_DiagNL3L	HistCall3
stc_DiagNL4L	HistCall4
stc_DiagNL5L	HistCall5
stc_DiagNL6L	HistCall6
stc_DiagNL7L	HistCall7
stc_DiagRes	HistReserve
stc_SmBatterie	SmBatterie
stc_QuitDI	QuitDI
stc_QuitDO	QuitDO

DDC Suite Advanced – Overview

Example: The FBox "ManualOp" has 3 outputs to indicate if there is a FBox in manual mode. If you'd like to have one symbol if any output is high than you can use of course an "OR" gate – but you can improve the FBox itself.

In file "AddOn_DDC_GeneralManualInfo.src" we could type in:

```
A.DDC_Suite.Name.Manual equ f ; New symbol indicating any FBox under manual operation
        publ A.DDC_Suite.Name.Manual ; make it available in symbol editor system tab

        sth       out_Motoren          ;
        orh       out_Antriebe         ;
        orh       out_Regler           ;
        out       A.DDC_Suite.Name.Manual ;
```

After a rebuild you'll see the new symbol in system tab an can use it in your program. This example is like "adding an FBox output"

In other cases it could be possible to get the real symbol name (you'll see in the IN or OUT connector in Fupla or the symbols connected to the FBox e.g. to use them (for mapping) in the BACnet configuration.

Therefore you have to use the syntax

@&SYMBOL@

This sequence will display the symbolname it self, e.g. if you add to the fomer example the line


```
$report The FBox adjust parameter @&stc_Regler@ contains the number of controllers under manual operation
```

Family : Initialisation

DDC Suite Advanced – Family: Initialisation
Init LIB

BACnet: BAC_DDC_InitLibrary.src
Doc-File: DOC_DDC_InitialisationLibrary.src
AddOn: AddOn_DDC_InitialisationLibrary.src

Label	Symbol
AckAla	in_QuitStoerung
AckMt	in_QuitWartung
WD	in_WatchDog
OutTemp	in_Aussentemp

Init LIB

AckAla ResAla
AckMt ResMt
WD Err
OutTemp Bat
Pulse

Label	Symbol
ResAla	out_ResetStoerung
ResMt	out_ResetWartung
Err	out_Error
Bat	out_Batterie
Pulse	out_SekundenPuls

Symbol	Default Symbol
stc_QuitStoerung	SmQuit
stc_QuitWartung	WartQuit
stc_QuitDiagnose	HistQuit
stc_DiagnoseSm	HistFehler
stc_ResetStoerungPuls	SmQuitHwImp
stc_DiagXob	HistMeldung
stc_DiagPrgL	HistProgLine
stc_DiagIdx	HistIndex
stc_DiagCobL	HistCobLine
stc_DiagNL1L	HistCall1
stc_DiagNL2L	HistCall2
stc_DiagNL3L	HistCall3
stc_DiagNL4L	HistCall4
stc_DiagNL5L	HistCall5
stc_DiagNL6L	HistCall6
stc_DiagNL7L	HistCall7
stc_DiagRes	HistReserve
stc_SmBatterie	SmBatterie
stc_QuitDI	QuitDI
stc_QuitDO	QuitDO

DDC Suite Advanced – Family: Initialisation
Init ABS

BACnet: BAC_DDC_InitAntiBlocking.src
Doc-File: DOC_DDC_InitialisationAntiBlocking.src
AddOn: AddOn_DDC_InitialisationAntiBlocking.src

Label	Symbol
En2P	in_FreigabeUwp
EnY	in_FreigabeY

Init ABS

-En2P   2P-
-EnY   Y-

Label	Symbol
2P	out_AbsUwp
Y	out_AbsY

Symbol	Default Symbol
stc_UwpFreigabe	UwpErlaubt
stc_UwpWochenpuls	UwpWoImpuls
stc_YFreigabe	YErlaubt
stc_YWochenpuls	YWoImpuls
stc_UwpVorwahl	UwpVorwahl
stc_UwpStillstand	UwpStillDauer
stc_UwpWochentag	UwpWoTag
stc_UwpUhrzeit	UwpWoUhrzeit
stc_YVorwahl	YVorwahl
stc_YStillstand	YStillDauer
stc_YWochentag	YWoTag
stc_YUhrzeit	YWoUhrzeit

DDC Suite Advanced – Family: Initialisation
Init FD

BACnet: BAC_DDC_InitFireDamper.src
Doc-File: DOC_DDC_InitialisationFireDamper.src
AddOn: AddOn_DDC_InitialisationFireDamper.src

Label	Symbol
Test	in_StartTest
Reset	in_Reset

Init FD

Test   Test
Reset   Date
              Time
              Fault
              Memory

Label	Symbol
Test	out_TestStatus
Date	out_TestDatum
Time	out_TestZeit
Fault	out_TestErgebnis
Memory	out_TestBeendet

Symbol	Default Symbol
stc_TestErgebnis	TestErg
stc_TestStarten	Start
stc_TestStoppen	Abbruch
stc_TestStatus	Zustand
stc_LaufzeitAuf	ZeitZu
stc_LaufzeitZu	ZeitAuf
stc_AnzahlBSK	BskTotal
stc_AnzahlPositiv	BskOk
stc_AnzahlNegativ	BskFehler
stc_TestDatum	TestDatum
stc_TestZeit	TestZeit

Family : General

DDC Suite Advanced – Family: General ManualOP

Doc-File: DOC_DDC_GeneralManualInfo.src
AddOn: AddOn_DDC_GeneralManualInfo.src

Label	Symbol
Motors	out_Motoren
Drives	out_Antriebe
Controll	out_Regler

Symbol	Default Symbol
stc_Motoren	MotAnzahl
stc_Antriebe	AntrAnzahl
stc_Regler	RegAnzahl
stc_MotorenAuto	MotAuto
stc_AntriebeAuto	AntrAuto
stc_ReglerAuto	RegAuto

Family : Analogue values

DDC Suite Advanced – Family: Analogue values
Value (binary)

BACnet: BAC_DDC_MeasurementBinary.src
Doc-File: DOC_DDC_MeasurementBinary.src

Label	Symbol
Value	in_Value

Value

Label	Symbol
	out_Value

Symbol	Default Symbol
stc_Value	Flag

DDC Suite Advanced – Family: Analogue values
Value (integer)

BACnet: BAC_DDC_MeasurementInteger.src
Doc-File: DOC_DDC_MeasurementInteger.src

Label	Symbol
Value	in_Value

Value

Label	Symbol
	out_Value

Symbol	Default Symbol
stc_Value	Register

DDC Suite Advanced – Family: Analogue values Sensor

Alarming: ALM_DDC_Measurement_Sensor.src
BACnet: BAC_DDC_MeasurementSensor.src
Doc-File: DOC_DDC_MeasurementSensor.src

Label	Symbol
Inp	in_Messwert

Sensor
- Inp   Value -
         AlaHigh -
         AlaLow -

Label	Symbol
Value	out_Messwert
AlaHigh	out_SmGwOben
AlaLow	out_SmGwUnten

Symbol	Default Symbol
stc_SmGwOben	SmGwOben
stc_SmGwUnten	SmGwUnten
stc_Messwert	Istwert
stc_Korrektur	Korrektur
stc_FilterZeit	FilterZeit
stc_FilterFaktor	FilterFaktor
stc_MesswertY1	IstwertY1
stc_MesswertY2	IstwertY2
stc_RohwertX1	RohwertX1
stc_RohwertX2	RohwertX2
stc_GwOben	GwOben
stc_GwUnten	GwUnten
stc_SpgGrp	SpgGrp
stc_MesswertTyp	MessTyp

DDC Suite Advanced – Family: Analogue values Counter

BACnet: BAC_DDC_MeasurementCounter.src
Doc-File: DOC_DDC_MeasurementCounter.src

Label	Symbol
Imp	in_Impuls

Counter
- Imp
- Val
- Res
- LastVal
- LastRes

Label	Symbol
Val	out_Zaehlwert
Res	out_Restwert
LastVal	out_ZaehlwertAlt
LastRes	out_RestwertAlt

Symbol	Default Symbol
stc_ImpulsFaktor	ImpFaktor
stc_ReduktionsFaktor	RedFaktor
stc_Zaehlwert	Wert
stc_Restwert	Rest
stc_Monat	Monat
stc_Tag	MonatTag
stc_ZaehlwertAlt	SpeicherWert
stc_RestwertAlt	SpeicherRest

BACnet: BAC_DDC_MeasurementSelect.src
Doc-File: DOC_DDC_MeasurementSelect.src

Label	Symbol
Inp	in_Wert

Selection
- Inp0   max.
- Inp1   Avg
- Inp2   min.
          #1
          #2

Label	Symbol
max.	out_Max
Avg	out_Mittel
min.	out_Min
#1	out_Wahl1
#2	out_Wahl2

Symbol	Default Symbol
stc_MitWert1	Wert1Aktiv
stc_MitWert2	Wert2Aktiv
stc_MitWert3	Wert3Aktiv
stc_MitWert4	Wert4Aktiv
stc_MitWert5	Wert5Aktiv
stc_MitWert6	Wert6Aktiv
stc_MitWert7	Wert7Aktiv
stc_MitWert8	Wert8Aktiv
stc_MitWert9	Wert9Aktiv
stc_MitWert10	Wert10Aktiv
stc_Max	Maximal
stc_Mittel	Mittel
stc_Min	Minimal
stc_Wahl1	Auswahl1
stc_Wahl2	Auswahl2
stc_Wert1	Wert1
stc_Wert2	Wert2

DDC Suite Advanced – Family: Analogue values A/D

BACnet: BAC_DDC_MeasurementAnalogDigital.src
Doc-File: DOC_DDC_MeasurementAnalogDigital.src

Label	Symbol
A	in_Wert

A/D
A   D

Label	Symbol
D	out_Zustand

Symbol	Default Symbol
stc_Zustand	Wert
stc_GrenzwertOben	High
stc_GrenzwertUnten	Low

Family : Alarming

DDC Suite Advanced – Family: Alarming
Ala Fuse

Alarming: ALM_DDC_Alarming_PowerSupply.src
BACnet: BAC_DDC_AlarmingPowerSupply.src
Doc-File: DOC_DDC_AlarmingPowerSupply.src
AddOn: AddOn_DDC_AlarmingPowerSupply.src

Label	Symbol
230	in_230AC
24~	in_24AC
24=	in_24DC
Ph	in_Phasen
vltg.	in_Steuerkreis

Ala Fuse
- 230   GrpAla
- 24~   StartUp
- 24=   [230]
- Ph   [24~]
- vltg. [24=]
         [Ph]
         [Spg]

Label	Symbol
GrpAla	out_Ssm
StartUp	out_StartUp
[230]	out_DI230AC
[24~]	out_DI24AC
[24=]	out_DI24DC
[Ph]	out_DIPhasen
[Spg]	out_DISteuerkreis

Symbol	Default Symbol
stc_NoNc230AC	AC230NoNc
stc_QuitTyp230AC	AC230QuitPflicht
stc_NoNc24AC	AC24NoNc
stc_QuitTyp24AC	AC24QuitPflicht
stc_NoNc24DC	DC24NoNc
stc_QuitTyp24DC	DC24QuitPflicht
stc_NoNcPhasen	PhasenNoNc
stc_QuitTypPhasen	PhasenQuitPflicht
stc_NoNcSteuerkreis	SpgNoNc
stc_QuitTypSteuerkreis	SpgQuitPflicht
stc_PhasenTyp	PhasenOverride
stc_SteuerkreisTyp	SpgOverride
stc_Sm230AC	SmAC230
stc_Sm24AC	SmAC24
stc_Sm24DC	SmDC24
stc_SmPhasen	SmPhasen
stc_SmSteuerkreis	SmSpg
stc_DI230AC	DI230AC
stc_DI24AC	DI24AC
stc_DI24DC	DI24DC
stc_DIPhasen	DIPhasen
stc_DISteuerkreis	DISteuerkreis
stc_Delay	Verzoeger

DDC Suite Advanced – Family: Alarming
1 Ala/Msg

Alarming: ALM_DDC_Alarming_AlarmMessage.src
BACnet: BAC_DDC_AlarmingAlarmMessage.src
Doc-File: DOC_DDC_AlarmingAlarmMessage.src
AddOn: AddOn_DDC_AlarmingAlarmMessage.src

Label	Symbol
Inp	in_Kontakt

1 Ala/Msg ⬤
Inp   Alarm
           Message
           [Inp]

Label	Symbol
Alarm	out_Sm
Message	out_Mld
[Inp]	out_DI

Symbol	Default Symbol
stc_NoNc	NoNc
stc_QuitTyp	QuitPflicht
stc_Sm	Sm
stc_SpgGrp	SpgGrp
stc_SmTyp	Funktion
stc_AnzugVerz	VerzAnzug
stc_AbfalVerz	VerzAbfall
stc_DI	DIKontakt

DDC Suite Advanced – Family: Alarming
1 Alarm

Alarming: ALM_DDC_Alarming_1Alarm.src
BACnet: BAC_DDC_Alarming1Alarm.src
Doc-File: DOC_DDC_Alarming1Alarm.src
AddOn: AddOn_DDC_Alarming1Alarm.src

Label	Symbol
Inp	in_Kontakt

1 Alarm ⬤

-Inp   Alarm-
         [Inp]-

Label	Symbol
Alarm	out_Sm
[Inp]	out_DI

Symbol	Default Symbol
stc_NoNc	NoNc
stc_QuitTyp	QuitPflicht
stc_SpgGrp	SpgGrp
stc_Sm	Sm
stc_DI	DIKontakt

DDC Suite Advanced – Family: Alarming
5 Alarms

Alarming: ALM_DDC_Alarming_5Alarms.src
BACnet: BAC_DDC_Alarming5Alarms.src
Doc-File: DOC_DDC_Alarming5Alarms.src
AddOn: AddOn_DDC_Alarming5Alarms.src

Label	Symbol
Inp	in_Kontakt

5 Alarms ⃝
- Inp0   GrpAla
- Inp1   Alarm0
          Alarm1
          [Inp]0
          [Inp]1

Label	Symbol
GrpAla	out_Ssm
Alarm	out_Sm
[Inp]	out_DI

Symbol	Default Symbol
stc_NoNc1	Mld1_NoNc
stc_QuitTyp1	Mld1_QuitPflicht
stc_NoNc2	Mld2_NoNc
stc_QuitTyp2	Mld2_QuitPflicht
stc_NoNc3	Mld3_NoNc
stc_QuitTyp3	Mld3_QuitPflicht
stc_NoNc4	Mld4_NoNc
stc_QuitTyp4	Mld4_QuitPflicht
stc_NoNc5	Mld5_NoNc
stc_QuitTyp5	Mld5_QuitPflicht
stc_SpgGrp1	Mld1_SpgGrp
stc_SpgGrp2	Mld2_SpgGrp
stc_SpgGrp3	Mld3_SpgGrp
stc_SpgGrp4	Mld4_SpgGrp
stc_SpgGrp5	Mld5_SpgGrp
stc_Sm1	Mld1_Sm
stc_Sm2	Mld2_Sm
stc_Sm3	Mld3_Sm
stc_Sm4	Mld4_Sm
stc_Sm5	Mld5_Sm
stc_DI1	Mld1_DIKontakt
stc_DI2	Mld2_DIKontakt
stc_DI3	Mld3_DIKontakt
stc_DI4	Mld4_DIKontakt
stc_DI5	Mld5_DIKontakt

DDC Suite Advanced – Family: Alarming
AlaDelay

Alarming: ALM_DDC_Alarming_DelayedAlarm.src
BACnet: BAC_DDC_AlarmingDelayedAlarm.src
Doc-File: DOC_DDC_AlarmingDelayedAlarm.src
AddOn: AddOn_DDC_AlarmingDelayedAlarm.src

Label	Symbol
En	in_Freigabe
Inp	in_Kontakt

AlaDelay ⬤
En   Alarm
Inp   [Inp]

Label	Symbol
Alarm	out_Sm
[Inp]	out_DI

Symbol	Default Symbol
stc_NoNc	NoNc
stc_VerzoegerungsT	VerzoegerTyp
stc_QuitTyp	QuitPflicht
stc_Dauer	Verzoegerung
stc_SpgGrp	SpgGrp
stc_Sm	Sm
stc_DI	DIKontakt

DDC Suite Advanced – Family: Alarming
Ala SetPt

Alarming: ALM_DDC_Alarming_Hysteresis.src
BACnet: BAC_DDC_AlarmingHysteresis.src
Doc-File: DOC_DDC_AlarmingHysteresis.src

Label	Symbol
En	in_Freigabe
SetPt	in_Sollwert
Sensor	in_Istwert

Ala SetPt
En   GrpAla
SetPt   TolHigh
Sensor   TolLow

Label	Symbol
GrpAla	out_Ssm
TolHigh	out_SmOben
TolLow	out_SmUnten

Symbol	Default Symbol
stc_Startverzoegerung	StartVerzoeger
stc_HystereseOben	UeberHyst
stc_VerzoegerungOben	UeberVerz
stc_HystereseUnten	UnterHyst
stc_VerzoegerungUnten	UnterVerz
stc_VerzoegerungEinheit	VerzProEinheit
stc_HystereseEinheit	SollAendHyst
stc_SmOben	UeberSm
stc_SmUnten	UnterSm
stc_Ssm	SmSammel
stc_Sollwert	Sollwert
stc_SollwertTyp	SollwertTyp

DDC Suite Advanced – Family: Alarming
AlaMotor1

Alarming: ALM_DDC_Alarming_MotorDrive1Speed.src
BACnet: BAC_DDC_AlarmingMotorDrive1Speed.src
Doc-File: DOC_DDC_AlarmingMotorDrive1Speed.src
AddOn: AddOn_DDC_AlarmingMotorDrive1Speed.src

Label	Symbol
Run	in_Ansteuerung
fb	in_Betrieb
mp	in_Motor
ss	in_RepSchalter
pfb	in_Prozess
man	in_Handeingriff

AlaMotor1 ⬤

Run	[fb!]
fb	[mp]
mp	[ss]
ss	[pfb]
pfb	[man]
man	GrpAla

Label	Symbol
[fb!]	out_DIBetrieb
[mp]	out_DIMotor
[ss]	out_DIRepSchalter
[pfb]	out_DIProzess
[man]	out_DIHandeingriff
GrpAla	out_Ssm

Symbol	Default Symbol
stc_ProzessNoNc	DrzNoNc
stc_MotorQuitTyp	MotQuitPflicht
stc_MotorNoNc	MotNoNc
stc_RepSchalterQuitTyp	RepQuitPflicht
stc_RepSchalterNoNc	RepNoNc
stc_HandeingriffNoNc	HandNoNc
stc_BetriebVerzoeger	BrmVerzoeger
stc_ProzessVerzoeger	DrzVerzoeger
stc_MotorSpgGrp	MotSpgGrp
stc_RepSchalterSpgGrp	RepSpgGrp
stc_HandeingriffSpgGrp	HandSpgGrp
stc_SmBetrieb	BrmSm
stc_SmProzess	DrzSm
stc_SmMotor	MotSm
stc_SmRepSchalter	RepSm
stc_SmHandeingriff	HandSm
stc_BetriebDI	BrmDI
stc_ProzessDI	DrzDI
stc_MotorDI	MotDI
stc_RepSchalterDI	RepDI
stc_HandeingriffDI	HandDI
stc_SsmTyp	SsmTyp

DDC Suite Advanced – Family: Alarming AlaMotor2

Alarming: ALM_DDC_Alarming_MotorDrive2Speed.src
BACnet: BAC_DDC_AlarmingMotorDrive2Speed.src
Doc-File: DOC_DDC_AlarmingMotorDrive2Speed.src
AddOn: AddOn_DDC_AlarmingMotorDrive2Speed.src

Label	Symbol
Run	in_Ansteuerung
fb1	in_BetriebSt1
fb2	in_BetriebSt2
mp1	in_MotorSt1
mp2	in_MotorSt2
ss	in_RepSchalter
pfb	in_Prozess
man1	in_HandeingriffSt1
man2	in_HandeingriffSt2

AlaMotor2

Run	[fb1!]
fb1	[fb2!]
fb2	[mp1]
mp1	[mp2]
mp2	[ss]
ss	[pfb]
pfb	[man1]
man1	[man2]
man2	GrpAla

Label	Symbol
[fb1!]	out_DIBetriebSt1
[fb2!]	out_DIBetriebSt2
[mp1]	out_DIMotorSt1
[mp2]	out_DIMotorSt2
[ss]	out_DIRepSchalter
[pfb]	out_DIProzess
[man1]	out_DIHandeingriffSt1
[man2]	out_DIHandeingriffSt2
GrpAla	out_Ssm

Symbol	Default Symbol
stc_ProzessNoNc	DrzNoNc
stc_MotorQuitTyp	MotQuitPflicht
stc_MotorNoNc	MotNoNc
stc_RepSchalterQuitTyp	RepQuitPflicht
stc_RepSchalterNoNc	RepNoNc
stc_HandeingriffNoNc	HandNoNc
stc_BetriebVerzoeger	BrmVerzoeger
stc_ProzessVerzoeger	DrzVerzoeger
stc_MotorSpgGrp	MotSpgGrp
stc_RepSchalterSpgGrp	RepSpgGrp
stc_HandeingriffSpgGrp	HandSpgGrp
stc_SmBetrieb	BrmSm
stc_SmProzess	DrzSm
stc_SmMotor	MotSm
stc_SmRepSchalter	RepSm
stc_SmHandeingriff	HandSm
stc_BetriebSt1DI	BrmSt1DI
stc_BetriebSt2DI	BrmSt2DI
stc_ProzessDI	DrzDI
stc_MotorSt1DI	MotSt1DI
stc_MotorSt2DI	MotSt2DI
stc_RepSchalterDI	RepDI
stc_HandeingriffSt1DI	HandSt1DI
stc_HandeingriffSt2DI	HandSt2DI
stc_SsmTyp	SsmTyp

DDC Suite Advanced – Family: Alarming AlaMotor3

Alarming: ALM_DDC_Alarming_MotorDrive3Speed.src
BACnet: BAC_DDC_AlarmingMotorDrive3Speed.src
Doc-File: DOC_DDC_AlarmingMotorDrive3Speed.src
AddOn: AddOn_DDC_AlarmingMotorDrive3Speed.src

Label	Symbol
Run	in_Ansteuerung
fb1	in_BetriebSt1
fb2	in_BetriebSt2
fb3	in_BetriebSt3
mp1	in_MotorSt1
mp2	in_MotorSt2
mp3	in_MotorSt3
ss	in_RepSchalter
pfb	in_Prozess
man1	in_HandeingriffSt1
man2	in_HandeingriffSt2
man3	in_HandeingriffSt3

AlaMotor3

Run [fb1!]
fb1 [fb2!]
fb2 [fb3!]
fb3 [mp1]
mp1 [mp2]
mp2 [mp3]
mp3 [ss]
ss [pfb]
pfb [man1]
man1 [man2]
man2 [man3]
man3 GrpAla

Label	Symbol
[fb1!]	out_DIBetriebSt1
[fb2!]	out_DIBetriebSt2
[fb3!]	out_DIBetriebSt3
[mp1]	out_DIMotorSt1
[mp2]	out_DIMotorSt2
[mp3]	out_DIMotorSt3
[ss]	out_DIRepSchalter
[pfb]	out_DIProzess
[man1]	out_DIHandeingriffSt1
[man2]	out_DIHandeingriffSt2
[man3]	out_DIHandeingriffSt3
GrpAla	out_Ssm

Symbol	Default Symbol
stc_ProzessNoNc	DrzNoNc
stc_MotorQuitTyp	MotQuitPflicht
stc_MotorNoNc	MotNoNc
stc_RepSchalterQuitTyp	RepQuitPflicht
stc_RepSchalterNoNc	RepNoNc
stc_HandeingriffNoNc	HandNoNc
stc_BetriebVerzoeger	BrmVerzoeger
stc_ProzessVerzoeger	DrzVerzoeger
stc_MotorSpgGrp	MotSpgGrp
stc_RepSchalterSpgGrp	RepSpgGrp
stc_HandeingriffSpgGrp	HandSpgGrp
stc_SmBetrieb	BrmSm
stc_SmProzess	DrzSm
stc_SmMotor	MotSm
stc_SmRepSchalter	RepSm
stc_SmHandeingriff	HandSm
stc_BetriebSt1DI	BrmSt1DI
stc_BetriebSt2DI	BrmSt2DI
stc_BetriebSt3DI	BrmSt3DI
stc_ProzessDI	DrzDI
stc_MotorSt1DI	MotSt1DI
stc_MotorSt2DI	MotSt2DI
stc_MotorSt3DI	MotSt3DI
stc_RepSchalterDI	RepDI
stc_HandeingriffSt1DI	HandSt1DI
stc_HandeingriffSt2DI	HandSt2DI
stc_HandeingriffSt3DI	HandSt3DI
stc_SsmTyp	SsmTyp

DDC Suite Advanced – Family: Alarming
Ala Frost

Alarming: ALM_DDC_Alarming_FrostProtection.src
BACnet: BAC_DDC_AlarmingFrostProtection.src
Doc-File: DOC_DDC_AlarmingFrostProtection.src
AddOn: AddOn_DDC_AlarmingFrostProtection.src

Label	Symbol
AirFt	in_Thermostat
RetT	in_Ruecklauftemp

Ala Frost

-AirFt   [AirFt]-
-RetT   AlaAirFt-
            AlaRetT-
            AlaFr-
            QitAla-

Label	Symbol
[AirFt]	out_DIThermostat
AlaAirFt	out_SmThermostat
AlaRetT	out_SmRuecklaufter
AlaFr	out_SmFrost
QitAla	out_Reset

Symbol	Default Symbol
stc_ThermostatNoNc	ThermNoNc
stc_RuecklauftempGwEin	RITempGwFrost
stc_Verzoegerung1te	Verzoeger1
stc_Verzoegerung2te	Verzoeger2
stc_ThermostatSpgGrp	ThermSpgGrp
stc_RuecklauftempGwAus	RITempGwOK
stc_SmThermostat	SmTherm
stc_SmRuecklauftemp	SmRITemp
stc_SmFrostprogramm	SmAktiv
stc_SmFrost	SmFrost
stc_ThermostatDI	ThermDI

DDC Suite Advanced – Family: Alarming FireDamper

Alarming: ALM_DDC_Alarming_FireDamper.src
BACnet: BAC_DDC_AlarmingFireDamper.src
Doc-File: DOC_DDC_AlarmingFireDamper.src
AddOn: AddOn_DDC_AlarmingFireDamper.src

Label	Symbol
Open	in_AnforderungAuf
fbC	in_RueckmeldungZu
fbO	in_RueckmeldungAuf
Test	in_Test

FireDamper ⬤

- Open   Open
- fbC   GrpAla
- fbO   AlaC
- Test   AlaO
                GrpMsg
                Test
                [fbO!]
                [fbC!]

Label	Symbol
Open	out_Auf
GrpAla	out_Ssm
AlaC	out_SmZu
AlaO	out_SmAuf
GrpMsg	out_Mld
Test	out_Test
[fbO!]	out_DIRmAuf
[fbC!]	out_DIRmZu

Symbol	Default Symbol
stc_QuitTyp	QuitLog
stc_SmZu	SmZu
stc_SmAuf	SmAuf
stc_AnsteuerungAuf	Betrieb
stc_RueckmeldungZu	RmZu
stc_RueckmeldungAuf	RmAuf
stc_Vorwahl	Vorwahl
stc_SmTyp	FunkOut
stc_ZuZeit	DauerZu
stc_AufZeit	DauerAuf
stc_RmAufDI	RmAufDI
stc_RmZuDI	RmZuDI

DDC Suite Advanced

Family : Set points

DDC Suite Advanced – Family: Set points
Val (binary)

BACnet: BAC_DDC_SetPointBinary.src
Doc-File: DOC_DDC_SetPointBinary.src

DDC Suite Advanced – Family: Set points
Val (integer)

BACnet: BAC_DDC_SetPointInteger.src
Doc-File: DOC_DDC_SetPointInteger.src

Label	Symbol
Val	out_Value

Symbol	Default Symbol
stc_Value	Register

DDC Suite Advanced – Family: Set points Hysteresis

BACnet: BAC_DDC_SetPointWithHysteresis.src
Doc-File: DOC_DDC_SetPointWithHysteresis.src

Label	Symbol
Ext.	in_Extern

Hysteresis

Ext.	+Hyst
	SetPt
	-Hyst

Label	Symbol
+Hyst	out_SollwertPlus
SetPt	out_Sollwert
-Hyst	out_SollwertMinus

Symbol	Default Symbol
stc_Basis	Basis
stc_Sollwert	SollMittel
stc_Hysterese	Hysterese
stc_SollwertPlus	SollPlus
stc_SollwertMinus	SollMinus
stc_TypExtern	ExtFunc
stc_SollwertExtern	Extern

DDC Suite Advanced – Family: Set points Linear2

BACnet: BAC_DDC_SetPointLinear2Point.src
Doc-File: DOC_DDC_SetPointLinear2Point.src

Label	Symbol
X	in_X

Linear2
X   Y

Label	Symbol
Y	out_Y

Symbol	Default Symbol
stc_X1	KurveX1
stc_Y1	KurveY1
stc_X2	KurveX2
stc_Y2	KurveY2
stc_Y	Errechnet

DDC Suite Advanced – Family: Set points Linear4

BACnet: BAC_DDC_SetPointLinear4Point.src
Doc-File: DOC_DDC_SetPointLinear4Point.src

Label	Symbol
X	in_X

Linear4

Label	Symbol
Y	out_Y

Symbol	Default Symbol
stc_X1	KurveX1
stc_X2	KurveX2
stc_X3	KurveX3
stc_X4	KurveX4
stc_Y1	KurveY1
stc_Y2	KurveY2
stc_Y3	KurveY3
stc_Y4	KurveY4
stc_Y	Errechnet

DDC Suite Advanced – Family: Set points Room

BACnet: BAC_DDC_SetPointRoom.src
Doc-File: DOC_DDC_SetPointRoom.src

Label	Symbol
Ext	in_Extern
OutTemp	in_Aussentemp

```
        Room
-Ext        SetPt-
-OutTemp
```

Label	Symbol
SetPt	out_Sollwert

Symbol	Default Symbol
stc_SollwertTyp	SollTyp
stc_SollwertBasis	Basis
stc_DifferenzDIN	DiffDin
stc_SollwertMaximal	Maximal
stc_ExternTyp	ExtFunc
stc_SollwertExtern	Extern
stc_Sollwert	Sollwert

Family : System and clocks

DDC Suite Advanced – Family: System and clocks
System 1sp

BACnet: BAC_DDC_EnablingSystemSwitch1.src
Doc-File: DOC_DDC_EnablingSystemSwitch1.src

Label	Symbol
Clock	in_Uhr
Special	in_Sonder
Channel	in_Kanal

System 1sp ⬤
-Clock   En-
-Special Demand-
-Channel

Label	Symbol
En	out_Freigabe
Demand	out_Bedarf

Symbol	Default Symbol
stc_UhrTyp	UhrTyp
stc_Uhr	Uhrkanal
stc_Bedarf	Bedarf
stc_Sonder	Sonder
stc_Freigabe	Freigabe
stc_Vorwahl	Vorwahl
stc_Schalter	VorwahlHand
stc_Kanal	Schaltkanal
stc_LED	

DDC Suite Advanced – Family: System and clocks
System 2sp

BACnet: BAC_DDC_EnablingSystemSwitch2.src
Doc-File: DOC_DDC_EnablingSystemSwitch2.src

Label	Symbol
ClkSp1	in_UhrSt1
ClkSp2	in_UhrSt2
SpcSp1	in_SonderSt1
SpcSp2	in_SonderSt2
Channel	in_Kanal

System 2sp
- ClkSp1   EnSp1
- ClkSp2   EnSp2
- SpcSp1 Demand
- SpcSp2
- Channel

Label	Symbol
EnSp1	out_Freigabe
EnSp2	out_FreigabeSt2
Demand	out_Bedarf

Symbol	Default Symbol
stc_UhrTyp	UhrTyp
stc_Uhr	UhrZustand
stc_Bedarf	Bedarf
stc_Sonder	Sonder
stc_Freigabe	Freigabe
stc_Vorwahl	Vorwahl
stc_Schalter	VorwahlHand
stc_Kanal	Schaltkanal

DDC Suite Advanced – Family: System and clocks
System 3sp

BACnet: BAC_DDC_EnablingSystemSwitch3.src
Doc-File: DOC_DDC_EnablingSystemSwitch3.src

Label	Symbol
ClkSp1	in_UhrSt1
ClkSp2	in_UhrSt2
ClkSp3	in_UhrSt3
SpcSp1	in_SonderSt1
SpcSp2	in_SonderSt2
SpcSp3	in_SonderSt3
Channel	in_Kanal

System 3sp
- ClkSp1   EnSp1
- ClkSp2   EnSp2
- ClkSp3   EnSp3
- SpcSp1 Demand
- SpcSp2
- SpcSp3
- Channel

Label	Symbol
EnSp1	out_Freigabe
EnSp2	out_FreigabeSt2
EnSp3	out_FreigabeSt3
Demand	out_Bedarf

Symbol	Default Symbol
stc_UhrTyp	UhrTyp
stc_Uhr	Uhrkanal
stc_Bedarf	Bedarf
stc_Sonder	Sonder
stc_Freigabe	Freigabe
stc_Vorwahl	Vorwahl
stc_Schalter	VorwahlHand
stc_Kanal	Schaltkanal

DDC Suite Advanced – Family: System and clocks HeatCirc

BACnet: BAC_DDC_EnablingSystemHeatingCirquit.src
Doc-File: DOC_DDC_EnablingSystemHeatingCirquit.src

Label	Symbol
Winter	in_Freigabe
Day	in_Tag
OutTemp	in_Aussentemp
FrostTmp	in_FrostTemp
Off	in_Vorrang
Channel	in_Kanal

HeatCirc
- Winter   En
- Day   Day
- OutTemp   SetPt
- FrostTmp
- Off
- Channel

Label	Symbol
En	out_Heizkreis
Day	out_Tag
SetPt	out_Sollwert

Symbol	Default Symbol
stc_Schalter	VorwahlHand
stc_Vorwahl	Vorwahl
stc_NachtAbs	NachtAbs
stc_HeizGwTag	HeizGwTag
stc_FrostGwEin	FrostGwEin
stc_FrostGwAus	FrostGwAus
stc_FbDauer	FbDauer
stc_Anforderung	Anforder
stc_Zustand	Zustand
stc_Sollwert	Sollwert
stc_SollwertAt	Errechnet
stc_KurveX1	KurveX1
stc_KurveY1	KurveY1
stc_KurveX2	KurveX2
stc_KurveY2	KurveY2
stc_KurveX3	KurveX3
stc_KurveY3	KurveY3
stc_KurveX4	KurveX4
stc_KurveY4	KurveY4
stc_Kanal	Schaltkanal
stc_HeizGwNacht	HeizGwNacht
stc_UhrTyp	UhrTyp
stc_Uhr	Uhr
stc_AtGwHyst	AtGwHyst

DDC Suite Advanced – Family: System and clocks HotWater

BACnet: BAC_DDC_EnablingSystemWaterHeater.src
Doc-File: DOC_DDC_EnablingSystemWaterHeater.src

Label	Symbol
Clock	in_Uhr
Ttop	in_TempOben
Tbot	in_TempUnten
Ala	in_Ssm
Channel	in_Kanal

HotWater

- Clock   En
- Ttop   Disinfec
- Tbot   SetPtInf
- Ala   SetPtHw
- Channel   TempHw
                Priority

Label	Symbol
En	out_Freigabe
Disinfec	out_LegFreigabe
SetPtInf	out_LadeSollwert
SetPtHw	out_Sollwert
TempHw	out_TempMittel
Priority	out_VorFreigabe

Symbol	Default Symbol
stc_Freigabe	Freigabe
stc_LegFreigabe	LegFreigabe
stc_VorVorwahl	VorVorwahl
stc_VorFreigabe	VorFreigabe
stc_Vorwahl	Vorwahl
stc_GwEin	GwEin
stc_GwAus	GwAus
stc_LadeHysterese	HystLadeTemp
stc_LadeSollwert	SwLadeTemp
stc_LegVorwahl	LegVorwahl
stc_LegWochentag	LegWoTag
stc_LegZeit	LegUhrzeit
stc_LegGwAus	LegGwAus
stc_LegDauer	LegMaxDauer
stc_VorDauer	VorMaxDauer
stc_Schalter	VorwahlHand
stc_Kanal	Kanal
stc_UhrTyp	UhrTyp
stc_Uhr	Uhr
stc_SsmSperre	Sperre

DDC Suite Advanced – Family: System and clocks
Boiler

BACnet: BAC_DDC_EnablingSystemBoiler.src
Doc-File: DOC_DDC_EnablingSystemBoiler.src

Label	Symbol
Winter	in_Freigabe
OutTemp	in_AussenTemp
SetPtCon	in_Sollwert

Boiler
-Winter   En
-OutTemp   SetPt
-SetPtCon

Label	Symbol
En	out_Freigabe
SetPt	out_Sollwert

Symbol	Default Symbol
stc_Heizen	Heizperiode
stc_Freigabe	Freigabe
stc_KurveX1	KurveX1
stc_KurveX2	KurveX2
stc_KurveX3	KurveX3
stc_KurveX4	KurveX4
stc_KurveY1	KurveY1
stc_KurveY2	KurveY2
stc_KurveY3	KurveY3
stc_KurveY4	KurveY4
stc_Schalter	VorwahlHand
stc_Vorwahl	Vorwahl
stc_Heizgrenze	Heizgrenze
stc_SollwertVerbr	SollVerbraucher
stc_SollwertAt	SollAt
stc_SollwertHyst	SollErhoehung
stc_Sollwert	Sollwert
stc_LED	

DDC Suite Advanced – Family: System and clocks FanStart

BACnet: BAC_DDC_EnablingStartAirCondition.src
Doc-File: DOC_DDC_EnablingStartAirCondition.src

Label	Symbol
Start	in_Freigabe
fbDamper	in_RmKlappeAuf
Ala	in_Stoerung
PreHReT	in_VeRITemp
AlaFrost	in_SmFrost
Dehumid	in_Entfeuchtung
SNCool	in_Sommernachtkuehlung
FrstProt	in_Auskuehlschutz
FrcdCool	in_Zwangskuehlung
SplyTemp	in_IstZuluftTemp

Fan Start

Start	WarmUp
fbDamper	Damper
Ala	SplyFan
PreHReT	ExhstFan
AlaFrost	Contr
Dehumid	SplyRamp
SNCool	SetPtMon
FrstProt	SplyStPt
FrcdCool	
SplyTemp	

Label	Symbol
WarmUp	out_Spuelen
Damper	out_Klappen
SplyFan	out_Zuluft
ExhstFan	out_Abluft
Contr	out_Regler
SplyRamp	out_SwZuluftRampe
SetPtMon	out_Grenzen
SplyStPt	out_SolZuluftTemp

Symbol	Default Symbol
stc_Spuelen	Vorspuehlung
stc_Klappen	Klappen
stc_Zuluft	Zuluefter
stc_Abluft	Abluefter
stc_Regler	Regelung
stc_Grenzen	Grenzwerte
stc_SpuelenDauer	VorspDauer
stc_AbluftVerz	AbluftVerz
stc_ReglerVerz	ReglerVerz
stc_GrenzenVerz	GwVerz
stc_SpuelenAt	GwAt
stc_SpuelenRIGw	GwRITemp
stc_SwZuluftRampe	ZulRampe
stc_SwRampe	ZulVerz
stc_SwZuluftMin	ZulMin
stc_SwZuluftTemp	ZulSoll

DDC Suite Advanced – Family: System and clocks FanSpecial

BACnet: BAC_DDC_EnablingAirConditionPreservation.src
Doc-File: DOC_DDC_EnablingAirConditionPreservation.src

Label	Symbol
Demand	in_Bedarf
EnSNCool	in_FrkAnfUhr
Value	in_RaumTempIst
SetPoint	in_RaumTempSoll

FanSpecial

- Demand   SNCool -
- EnSNCool   FrstProt -
- Value   FrcdCool -
- SetPoint   SpcSp1 -
    SpcSp2 -
    SpcSp3 -

Label	Symbol
SNCool	out_Sommernachtkuehlung
FrstProt	out_Auskuehlschutz
FrcdCool	out_Zwangskuehlung
SpcSp1	out_Stufe1
SpcSp2	out_Stufe2
SpcSp3	out_Stufe3

Symbol	Default Symbol
stc_FrkAt	FrkAtFreigabe
stc_FrkRt	FrkRtFreigabe
stc_FrkFreigabe	FrkFreigabe
stc_AksFreigabe	AksFreigabe
stc_UehsFreigabe	UehsFreigabe
stc_FrkVorwahl	FrkVorwahl
stc_FrkAtMin	FrkAtMin
stc_FrkAtRtHystEin	FrkAtHystRtEin
stc_FrkAtRtHystAus	FrkAtHystRtAus
stc_FrkRtHyst	FrkHystRt
stc_AksVorwahl	AksVorwahl
stc_AksGwEin	AksGwEin
stc_AksGwAus	AksGwAus
stc_UehsVorwahl	UehsVorwahl
stc_UehsHystEin	UehsHystEin
stc_UehsHystAus	UehsHystAus
stc_FrkAnfUhr	FrkAnfUhr

DDC Suite Advanced – Family: System and clocks
FanSpeed

BACnet: BAC_DDC_EnablingAirConditionDemand2ndSpeed.src
Doc-File: DOC_DDC_EnablingAirConditionDemand2ndSpeed.src

Label	Symbol
En	in_Freigabe
ClkSp2	in_Uhr
SetPt	in_Sollwert
Value	in_Istwert

FanSpeed
- En   Sp2
- ClkSp2
- SetPt
- Value

Label	Symbol
Sp2	out_Stufe2

Symbol	Default Symbol
stc_Uhr	Uhr
stc_Last	Last
stc_Stufe2	Stufe2
stc_LastTyp	LastTyp
stc_HystereseEin	HystEin
stc_VerzoegerungEin	VerzEin
stc_HystereseAus	HystAus
stc_VerzoegerungAus	VerzAus

DDC Suite Advanced – Family: System and clocks Redundant

BACnet: BAC_DDC_EnablingRedundant.src
Doc-File: DOC_DDC_EnablingRedundant.src

Label	Symbol
En1	in_En1
En2	in_En2
Ala1	in_Sm1
Ala2	in_Sm2
CntH1	in_Std1
CntH2	in_Std2

```
        Redundant
  ┌──────────────────┐
 ─┤En1          Run1├─
 ─┤En2          Run2├─
 ─┤Ala1             │
 ─┤Ala2             │
 ─┤CntH1            │
 ─┤CntH2            │
  └──────────────────┘
```

Label	Symbol
Run1	out_Uwp1
Run2	out_Uwp2

Symbol	Default Symbol
stc_Vorwahl	Vorwahl
stc_Typ	Funktion
stc_StdDifferenz	Differenz
stc_Wochentag	WoTag
stc_Uhrzeit	Uhrzeit
stc_Invertieren	Invertieren
stc_Folge	Folge

Family : Controls

DDC Suite Advanced – Family: Controls
Motor 1

Alarming: ALM_DDC_Control_MotorDrive1Speed.src
BACnet: BAC_DDC_ControlMotorDrive1Speed.src
Doc-File: DOC_DDC_ControlMotorDrive1Speed.src
AddOn: AddOn_DDC_ControlMotorDrive1Speed.src

Label	Symbol
En	in_Anforderung
fb	in_Betrieb
Ala	in_SsmSperre

Motor 1
- En
- fb
- Ala
- Run
- CntH
- CntFb
- Mt

Label	Symbol
Run	out_Ausgang
CntH	out_Stunden
CntFb	out_Schaltungen
Mt	out_Wartung

Symbol	Default Symbol
stc_HMI	HMI
stc_Startverzoegerun	StartVerzoeger
stc_SchaltungenMax	SchaltungMax
stc_StundenMax	StundenMax
stc_Ansteuerung	Ansteuerung
stc_Betrieb	Betrieb
stc_Wartung	Wartung
stc_SsmSperre	Sperre
stc_Schaltungen	Schaltung
stc_Stunden	Stunden
stc_AnsteuerungDO	AnsteuerDO
stc_HMISuper	HMISuper
stc_Ausgang	Ausgang

DDC Suite Advanced – Family: Controls
Motor 2

Alarming: ALM_DDC_Control_MotorDrive2Speed.src
BACnet: BAC_DDC_ControlMotorDrive2Speed.src
Doc-File: DOC_DDC_ControlMotorDrive2Speed.src
AddOn: AddOn_DDC_ControlMotorDrive2Speed.src

Label	Symbol
En1	in_Anforderung
En2	in_AnforderungSt2
fb1	in_BetriebSt1
fb2	in_BetriebSt2
Ala	in_SsmSperre

Motor 2

En1   Run1
En2   Run2
fb1   CntH
fb2   CntFb
Ala   Mt

Label	Symbol
Run1	out_AnsteuerungSt1
Run2	out_AnsteuerungSt2
CntH	out_Stunden
CntFb	out_Schaltungen
Mt	out_Wartung

Symbol	Default Symbol
stc_HMI	HMI
stc_Startverzoegerung	StartVerzoeger
stc_VerzoegerungSt2	VerzoegerSt2
stc_SchaltungenMax	SchaltungMax
stc_StundenMax	StundenMax
stc_VerzoegerungSt1	VerzoegerSt1
stc_Wartung	Wartung
stc_SsmSperre	Sperre
stc_Ansteuerung	Ansteuerung
stc_Betrieb	Betrieb
stc_SchaltungenSt1	SchaltungSt1
stc_StundenSt1	StundenSt1
stc_SchaltungenSt2	SchaltungSt2
stc_StundenSt2	StundenSt2
stc_AnsteuerungSt1DO	AnsteuerSt1DO
stc_AnsteuerungSt2DO	AnsteuerSt2DO
stc_HMISuper	HMISuper

DDC Suite Advanced – Family: Controls
Motor 3

Alarming: ALM_DDC_Control_MotorDrive3Speed.src
BACnet: BAC_DDC_ControlMotorDrive23peed.src
Doc-File: DOC_DDC_ControlMotorDrive23peed.src
AddOn: AddOn_DDC_ControlMotorDrive3Speed.src

Label	Symbol
En1	in_Anforderung
En2	in_AnforderungSt2
En3	in_AnforderungSt3
fb1	in_BetriebSt1
fb2	in_BetriebSt2
fb3	in_BetriebSt3
Ala	in_SsmSperre

Motor 3

En1	Run1
En2	Run2
En3	Run3
fb1	CntH
fb2	CntFb
fb3	Mt
Ala	

Label	Symbol
Run1	out_AnsteuerungSt1
Run2	out_AnsteuerungSt2
Run3	out_AnsteuerungSt3
CntH	out_Stunden
CntFb	out_Schaltungen
Mt	out_Wartung

Symbol	Default Symbol
stc_HMI	HMI
stc_Startverzoegerung	StartVerzoeger
stc_VerzoegerungSt2	VerzoegerSt2
stc_SchaltungenMax	SchaltungMax
stc_StundenMax	StundenMax
stc_VerzoegerungSt1	VerzoegerSt1
stc_Wartung	Wartung
stc_SsmSperre	Sperre
stc_Ansteuerung	Ansteuerung
stc_Betrieb	Betrieb
stc_SchaltungenSt1	SchaltungSt1
stc_StundenSt1	StundenSt1
stc_SchaltungenSt2	SchaltungSt2
stc_StundenSt2	StundenSt2
stc_SchaltungenSt3	SchaltungSt3
stc_StundenSt3	StundenSt3
stc_AnsteuerungSt1DO	AnsteuerSt1DO
stc_AnsteuerungSt2DO	AnsteuerSt2DO
stc_AnsteuerungSt3DO	AnsteuerSt3DO
stc_HMISuper	HMISuper

DDC Suite Advanced – Family: Controls Pump

Alarming: ALM_DDC_Control_Pump.src
BACnet: BAC_DDC_ControlPump.src
Doc-File: DOC_DDC_ControlPump.src
AddOn: AddOn_DDC_ControlPump.src

Label	Symbol
En	in_Anforderung
fb	in_Betrieb
Ala	in_Ssm
Y	in_Y

Pump
- En   Run
- fb   CntH
- Ala   CntFb
- Y   Mt

Label	Symbol
Run	out_Ausgang
CntH	out_Stunden
CntFb	out_Schaltungen
Mt	out_Wartung

Symbol	Default Symbol
stc_HMI	HMI
stc_Nachlauf	Nachlauf
stc_AbsVorwahl	AbsErlaubt
stc_SchaltungenMax	SchaltungMax
stc_StundenMax	StundenMax
stc_AtFunktion	BedAtFunk
stc_AtGrenzwert	BedAtGw
stc_YFunktion	BedYFunk
stc_YGrenzwert	BedYGw
stc_Ansteuerung	Ansteuerung
stc_Betrieb	Betrieb
stc_Wartung	Wartung
stc_SsmSperre	Sperre
stc_AtAnforderung	BedAt
stc_YAnforderung	BedY
stc_Schaltungen	Schaltung
stc_Stunden	Stunden
stc_AnsteuerungDO	AnsteuerDO
stc_HMISuper	HMISuper
stc_Ausgang	Ausgang

DDC Suite Advanced – Family: Controls Continous

Alarming: ALM_DDC_Control_ValveDamperAnalog.src
BACnet: BAC_DDC_ControlValveDamperAnalog.src
Doc-File: DOC_DDC_ControlValveDamperAnalog.src
AddOn: AddOn_DDC_ControlValveDamperAnalog.src

Label	Symbol
Y	in_Signal
Yfb	in_Rueckmeldung
Ovr	in_Zwangssteuerung
man	in_Handeingriff

Continous

- Y — Y
- Yfb — Yinv
- Ovr — GrpAla
- man — AlaHigh
- AlaLow
- [man]

Label	Symbol
Y	out_Signal
Yinv	out_SignalInvers
GrpAla	out_Ssm
AlaHigh	out_SmOben
AlaLow	out_SmUnten
[man]	out_Handeingriff

Symbol	Default Symbol
stc_Antiblock	AbsErlaubt
stc_Laufzeit	Laufzeit
stc_SignalHand	SignalHand
stc_SignalZwang	SignalZwang
stc_SignalRegler	SignalRegler
stc_Signal	Signal
stc_SignalInvers	SignalInvers
stc_RueckmeldungHyst	RmHyst
stc_RueckmeldungSpgGrp	RmSpgGrp
stc_RueckmeldungRohMin	RmRohMin
stc_RueckmeldungRohMax	RmRohMax
stc_Rueckmeldung	RmSignal
stc_HMI	Vorwahl
stc_Zwangssteuerung	ZwangAktiv
stc_Invertieren	Invertieren
stc_RueckmeldungQuit	RmQuitPflicht
stc_RueckmeldungSmOben	RmObenSm
stc_RueckmeldungSmUnten	RmUntenSm
stc_HandeingriffDI	HandDI
stc_HandeingriffNoNc	HandNoNc
stc_HandeingriffSpgGrp	HandSpg
stc_HandeingriffSm	HandSm
stc_SignalAnforderung	Anforder

DDC Suite Advanced – Family: Controls
O/C

Alarming: ALM_DDC_Control_ValveDamperOpenClose.src
BACnet: BAC_DDC_ControlValveDamperOpenClose.src
Doc-File: DOC_DDC_ControlValveDamperOpenClose.src
AddOn: AddOn_DDC_ControlValveDamperOpenClose.src

Label	Symbol
En	in_AnforderungAuf
fbO	in_RueckmeldungAuf
fbC	in_RueckmeldungZu
Ovr	in_Zwangssteuerung
man	in_Handeingriff

O/C
En   Up
fbO   GrpAla
fbC   AlaO
Ovr   AlaC
man   [fbO!]
          [fbC!]
          [man]

Label	Symbol
Up	out_AusgangAuf
GrpAla	out_Ssm
AlaO	out_RueckmeldungAufSm
AlaC	out_RueckmeldungZuSm
[fbO!]	out_RueckmeldungAuf
[fbC!]	out_RueckmeldungZu
[man]	out_Handeingriff

Symbol	Default Symbol
stc_Antiblock	AbsErlaubt
stc_Laufzeit	Laufzeit
stc_RueckmeldungSpgGrp	RmSpgGrp
stc_HMI	Vorwahl
stc_Nachlauf	Nachlauf
stc_Zwangssteuerung	ZwangAktiv
stc_RueckmeldungQuit	RmQuitPflicht
stc_RueckmeldungAufSm	RmAufSm
stc_RueckmeldungZuSm	RmZuSm
stc_Zwangsbefehl	BetriebZwang
stc_AnsteuerungAuf	Betrieb
stc_Rueckmeldung	Rm
stc_RueckmeldungTyp	RmTyp
stc_RmAufDI	RmAufDI
stc_RmZuDI	RmZuDI
stc_HMISuper	HMISuper
stc_HandeingriffDI	HandDI
stc_HandeingriffNoNc	HandNoNc
stc_HandeingriffSpgGrp	HandSpg
stc_HandeingriffSm	HandSm
stc_AusgangAuf	Ausgang
stc_AnstDO	AnsteuerDO

DDC Suite Advanced

Family : Controller

DDC Suite Advanced – Family: Controller Master

BACnet: BAC_DDC_RegulationCascadeMaster.src
Doc-File: DOC_DDC_RegulationCascadeMaster.src

Label	Symbol
EnContr	in_FrgRegelung
Temp	in_Istwert
SetPt	in_Sollwert
SeqMin	in_YMin
SeqMax	in_YMax

Master
- EnContr SetPtSeq
- Temp SetPt
- SetPt
- SeqMin
- SeqMax

Label	Symbol
SetPtSeq	out_Y
SetPt	out_Sollwert

Symbol	Default Symbol
stc_SollwertWahl	SollwertVorwahl
stc_MinMaxWahl	SignalMMVorw
stc_YWahl	SignalHaVorw
stc_Auskuehlschutz	Meldung4
stc_Zwangskuehlung	Meldung5
stc_P_Band	PBand
stc_I_Zeit	Nachstell
stc_D_Anteil	Diffential
stc_Totzone	Totzone
stc_Abtasten	Abtastzeit
stc_YHand	SignalHand
stc_YMin	SignalMin
stc_YMax	SignalMax
stc_Sollwert	Sollwert
stc_Istwert	Istwert
stc_Y	Signal

DDC Suite Advanced – Family: Controller MasterZero

BACnet: BAC_DDC_RegulationCascadeMasterNull.src
Doc-File: DOC_DDC_RegulationCascadeMasterNull.src

Label	Symbol
EnContr	in_FrgRegelung
Temp	in_Istwert
SetPtC	in_SollwertKuehlen
SetPtH	in_SollwertHeizen
SeqMin	in_YMin
SeqMax	in_YMax

MasterZero

Label	Symbol
SeqSPC	out_YKuehlen
SeqSPH	out_YHeizen
SetPtC	out_SollwertKuehlen
SetPtH	out_SollwertHeizen

Symbol	Default Symbol
stc_SollwertWahl	SollwertVorwahl
stc_MinMaxWahl	SignalMMVorw
stc_YWahl	SignalHaVorw
stc_Auskuehlschutz	Meldung4
stc_Zwangskuehlung	Meldung5
stc_P_Band	PBand
stc_I_Zeit	Nachstell
stc_D_Anteil	Diffential
stc_Totzone	Totzone
stc_Abtasten	Abtastzeit
stc_YHandKuehlen	SignalHandKh
stc_YHandHeizen	SignalHandHz
stc_YMin	SignalMin
stc_YMax	SignalMax
stc_SollwertKuehlen	SollwertKh
stc_SollwertHeizen	SollwertHz
stc_Istwert	Istwert
stc_YKuehlen	SignalKh
stc_YHeizen	SignalHz

Title of this presentation

59

DDC Suite Advanced – Family: Controller Cooler

BACnet: BAC_DDC_RegulationCooler.src
Doc-File: DOC_DDC_RegulationCooler.src

Label	Symbol
EnContr	in_FrgRegelung
SetPt	in_Sollwert
Temp	in_Istwert
Lock	in_Sperre

Cooler
- EnContr Y
- SetPt Act
- Temp SetPt
- Lock
- C ????
- + ????

Label	Symbol
Y	out_Y
Act	out_ReglerAktiv
SetPt	out_Sollwert

Symbol	Default Symbol
stc_SollwertWahl	SollwertVorwahl
stc_MinMaxWahl	SignalMMVorw
stc_YWahl	SignalHaVorw
stc_Meldung4	Meldung4
stc_Meldung5	Meldung5
stc_P_Band	PBand
stc_I_Zeit	Nachstell
stc_D_Anteil	Diffential
stc_Totzone	Totzone
stc_Abtasten	Abtastzeit
stc_YHand	SignalHand
stc_YMin	SignalMin
stc_YMax	SignalMax
stc_Sollwert	Sollwert
stc_Istwert	Istwert
stc_Y	Signal

DDC Suite Advanced – Family: Controller HeatRec

BACnet: BAC_DDC_RegulationHeatRecovery.src
Doc-File: DOC_DDC_RegulationHeatRecovery.src

Label	Symbol
EnContr	in_FrgRegelung
SetPt	in_Sollwert
Temp	in_Istwert
RoomTemp	in_Raumtemp
Ala	in_SmWRG
YFrost	in_YFrost

HeatRec

- EnContr   Y-
- SetPt   Act-
- Temp   SetPt-
- RoomTemp
- Ala
- YFrost
- -   ????
- HR   ????
- +   ????

Label	Symbol
Y	out_Y
Act	out_ReglerAktiv
SetPt	out_Sollwert

Symbol	Default Symbol
stc_SollwertWahl	SollwertVorwahl
stc_MinMaxWahl	SignalMMVorw
stc_YWahl	SignalHaVorw
stc_Meldung4	Meldung4
stc_Meldung5	Meldung5
stc_P_Band	PBand
stc_I_Zeit	Nachstell
stc_D_Anteil	Diffential
stc_Totzone	Totzone
stc_Abtasten	Abtastzeit
stc_YHand	SignalHand
stc_YMin	SignalMin
stc_YMax	SignalMax
stc_Sollwert	Sollwert
stc_Istwert	Istwert
stc_Y	Signal

DDC Suite Advanced – Family: Controller MixedAir

BACnet: BAC_DDC_RegulationMixedAir.src
Doc-File: DOC_DDC_RegulationMixedAir.src

Label	Symbol
EnContr	in_FrgRegelung
SetPt	in_Sollwert
Temp	in_Istwert
RoomTemp	in_Raumtemp

MixedAir
- EnContr   Y
- SetPt   YInv
- Temp   Act
- RoomTemp   SetPt
-   ????
- MA ????
- + ????

Label	Symbol
Y	out_YMI
YInv	out_YFI
Act	out_ReglerAktiv
SetPt	out_Sollwert

Symbol	Default Symbol
stc_SollwertWahl	SollwertVorwahl
stc_MinMaxWahl	SignalMMVorw
stc_YWahl	SignalHaVorw
stc_Meldung4	Meldung4
stc_Meldung5	Meldung5
stc_P_Band	PBand
stc_I_Zeit	Nachstell
stc_D_Anteil	Diffential
stc_Totzone	Totzone
stc_Abtasten	Abtastzeit
stc_YHand	SignalHand
stc_YMin	SignalMin
stc_YMax	SignalMax
stc_Sollwert	Sollwert
stc_Istwert	Istwert
stc_Y	Signal

DDC Suite Advanced – Family: Controller Preheater

BACnet: BAC_DDC_RegulationPreheater.src
Doc-File: DOC_DDC_RegulationPreheater.src

Label	Symbol
EnContr	in_FrgRegelung
SetPt	in_Sollwert
Temp	in_Istwert
YCldStrt	in_YKaltstart

PreHeater
- EnContr   Y -
- SetPt   Act -
- Temp   SetPt -
- YCldStrt
- _   ????
- PH   ????
- +   ????

Label	Symbol
Y	out_Y
Act	out_ReglerAktiv
SetPt	out_Sollwert

Symbol	Default Symbol
stc_SollwertWahl	SollwertVorwahl
stc_MinMaxWahl	SignalMMVorw
stc_YWahl	SignalHaVorw
stc_Meldung4	Meldung4
stc_Meldung5	Meldung5
stc_P_Band	PBand
stc_I_Zeit	Nachstell
stc_D_Anteil	Diffential
stc_Totzone	Totzone
stc_Abtasten	Abtastzeit
stc_YHand	SignalHand
stc_YMin	SignalMin
stc_YMax	SignalMax
stc_Sollwert	Sollwert
stc_Istwert	Istwert
stc_Y	Signal

DDC Suite Advanced – Family: Controller ReHeater

BACnet: BAC_DDC_RegulationHeater.src
Doc-File: DOC_DDC_RegulationHeater.src

Label	Symbol
EnContr	in_FrgRegelung
SetPt	in_Sollwert
Temp	in_Istwert

ReHeater
- EnContr   Y -
- SetPt   Act -
- Temp   SetPt -
- ????
- RH ????

Label	Symbol
Y	out_Y
Act	out_ReglerAktiv
SetPt	out_Sollwert

Symbol	Default Symbol
stc_SollwertWahl	SollwertVorwahl
stc_MinMaxWahl	SignalMMVorw
stc_YWahl	SignalHaVorw
stc_Meldung4	Meldung4
stc_Meldung5	Meldung5
stc_P_Band	PBand
stc_I_Zeit	Nachstell
stc_D_Anteil	Diffential
stc_Totzone	Totzone
stc_Abtasten	Abtastzeit
stc_YHand	SignalHand
stc_YMin	SignalMin
stc_YMax	SignalMax
stc_Sollwert	Sollwert
stc_Istwert	Istwert
stc_Y	Signal

DDC Suite Advanced – Family: Controller Humid

BACnet: BAC_DDC_RegulationHumidifying.src
Doc-File: DOC_DDC_RegulationHumidifying.src

Label	Symbol
Encontr	in_FrgRegelung
SetPt	in_Sollwert
Humidity	in_Istwert

Humid
- Encontr   Y-
- SetPt   Act-
- Humidity   SetPt-
- Hum   ????
- DeH   ????

Label	Symbol
Y	out_Y
Act	out_ReglerAktiv
SetPt	out_Sollwert

Symbol	Default Symbol
stc_SollwertWahl	SollwertVorwahl
stc_MinMaxWahl	SignalMMVorw
stc_YWahl	SignalHaVorw
stc_Meldung4	Meldung4
stc_Meldung5	Meldung5
stc_P_Band	PBand
stc_I_Zeit	Nachstell
stc_D_Anteil	Diffential
stc_Totzone	Totzone
stc_Abtasten	Abtastzeit
stc_YHand	SignalHand
stc_YMin	SignalMin
stc_YMax	SignalMax
stc_Sollwert	Sollwert
stc_Istwert	Istwert
stc_Y	Signal

DDC Suite Advanced – Family: Controller DeHumid

BACnet: BAC_DDC_RegulationDehumidifying.src
Doc-File: DOC_DDC_RegulationDehumidifying.src

Label	Symbol
EnContr	in_FrgRegelung
SetPt	in_Sollwert
Humidity	in_Istwert
YCooler	in_YKuehler

DeHumid
- EnContr   Y
- SetPt   Act
- Humidity   Dehumid
- YCooler   SetPt
- Hum   ????
- DeH   ????

Label	Symbol
Y	out_Y
Act	out_ReglerAktiv
Dehumid	out_Entfeuchten
SetPt	out_Sollwert

Symbol	Default Symbol
stc_SollwertWahl	SollwertVorwahl
stc_MinMaxWahl	SignalMMVorw
stc_YWahl	SignalHaVorw
stc_Meldung4	Meldung4
stc_Meldung5	Meldung5
stc_P_Band	PBand
stc_I_Zeit	Nachstell
stc_D_Anteil	Diffential
stc_Totzone	Totzone
stc_Abtasten	Abtastzeit
stc_YHand	SignalHand
stc_YMin	SignalMin
stc_YMax	SignalMax
stc_Sollwert	Sollwert
stc_Istwert	Istwert
stc_Y	Signal

DDC Suite Advanced – Family: Controller
Controller

BACnet: BAC_DDC_RegulationController.src
Doc-File: DOC_DDC_RegulationController.src

Label	Symbol
En	in_FrgRegelung
Invers	in_Wirksinn
ContrVal	in_Istwert
SetPt	in_Sollwert
YMin	in_YMin
YMax	in_YMax

Controller
- En   Y
- Invers   SetPt
- ContrVal
- SetPt
- YMin
- YMax

Label	Symbol
Y	out_Y
SetPt	out_Sollwert

Symbol	Default Symbol
stc_SollwertWahl	SollwertVorwahl
stc_MinMaxWahl	SignalMMVorw
stc_YWahl	SignalHaVorw
stc_Meldung4	Meldung4
stc_Meldung5	Meldung5
stc_P_Band	PBand
stc_I_Zeit	Nachstell
stc_D_Anteil	Diffential
stc_Totzone	Totzone
stc_Abtasten	Abtastzeit
stc_YHand	SignalHand
stc_YMin	SignalMin
stc_YMax	SignalMax
stc_Sollwert	Sollwert
stc_Istwert	Istwert
stc_Y	Signal

DDC Suite Advanced – Family: Controller Limiter

BACnet: BAC_DDC_RegulationLimitation.src
Doc-File: DOC_DDC_RegulationLimitation.src

Label	Symbol
ContrVal	in_Istwert
SetPt	in_Sollwert

Limiter

ContrVal Y
SetPt SetPt

Label	Symbol
Y	out_Y
SetPt	out_Sollwert

Symbol	Default Symbol
stc_SollwertWahl	SollwertVorwahl
stc_MinMaxWahl	SignalMMVorw
stc_YWahl	SignalHaVorw
stc_Meldung4	Meldung4
stc_Meldung5	Meldung5
stc_P_Band	PBand
stc_I_Zeit	Nachstell
stc_D_Anteil	Diffential
stc_Totzone	Totzone
stc_Abtasten	Abtastzeit
stc_YHand	SignalHand
stc_YMin	SignalMin
stc_YMax	SignalMax
stc_Sollwert	Sollwert
stc_Istwert	Istwert
stc_Y	Signal

DDC Suite – hints & tricks

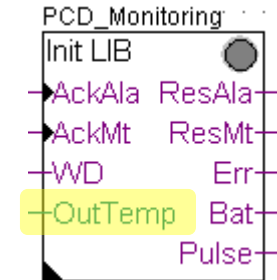**Stephan Hintze / 05.12.2008**

Initialisation

DDC Suite Advanced – Overview

The FBox "InitLIB" supports the outdoor temperature at input "OutTemp" to FBoxes for calculation or comparison.

This means that all such FBoxes in program will use a single unique outdoor temperature. But what if each system has it's own outdoor temperature sensor?

After a first build the FBox "InitLIB" creates a symbol in system tab which can be used to assign in program another outdoor temperature. So it's easy to map for each system it's own temperature.
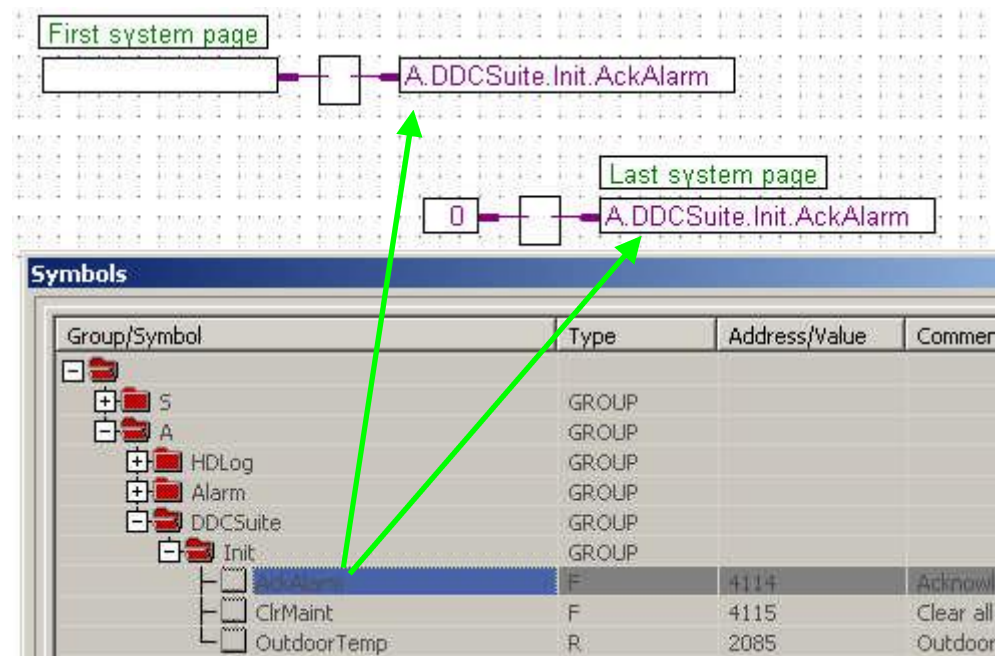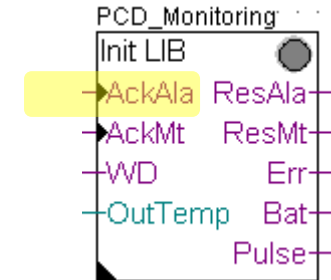
DDC Suite Advanced – Overview

Acknowledging alarms also is done from the FBox "InitLIB" for whole program. You can also split this into system parts. Just transfer the "acknowledge" command from system on first page of system into the provided symbol and reset it on last system page.

DDC Suite Advanced – Overview

Resetting maintenance messages and reinitializing internal counter is also done from the FBox "InitLIB" for whole program. You can also split this into system parts. Just transfer the "clear maintenance" command from system on first page of system into the provided symbol and reset it on last system page.