

# VIRTUAL COMMISSIONING OF A LARGE LNG PLANT WITH THE DCS „800XA“ BY ABB

**Herbert Krause**

Secolon, 32423 Minden, Schwarzer Weg 8, Germany

*herbert.krause@secolon.de (Herbert Krause)*

## **Abstract**

In Norway one of the largest Liquid Natural Gas (LNG) plants worldwide is under construction (Hammerfest, Natural gas field Snøhvit). The control and automation of the plant is done with the digital control system (DCS) “Industrial IT System 800xA” manufactured by ABB.

The complexity of the plant is very high and additionally some processes were newly introduced. That is why it was decided to perform the testing, training and virtual commissioning with a high fidelity simulator. The simulator used consists of a process simulator manufactured by Kongsberg Process Simulation, Sandvika (formerly Fantoft) and an emulator of the AC 870P / Melody system which is the active control part of the System 800xA. The original human system interface is used so that an overall test of the complete system is achieved.

One of the various challenges of this project is that the engineering, testing (virtual commissioning) and operator training is done simultaneously in different places in Norway: the engineering is done in the facilities in Bergen, testing is done in Oslo, training is performed in Hammerfest. Measures had to be taken to ensure the integrity of the data involved.

This affects the whole workflow. It had to be assured that diverging data may be harmonized again and that already achieved solutions do not get lost.

**Keywords: DCS, Emulation, Virtual commissioning**

## **Presenting Author’s biography**

Herbert Krause was born in Wilhelmshaven, Germany. He studied Mechanical Engineering at the Technical University of Braunschweig where he obtained his degree in 1982. He worked for a couple of years for Hartmann & Braun in the area of advanced control algorithms and the development of simulation and power plant management software. In 1995 he won two awards for papers dealing with the simulation and optimization of MSF desalination plants. In 2001 he developed the AC 870P / Melody Emulator. In 2006 he founded a company dealing with the development and support of high valued technical software.



## 1 The LNG Plant

Currently the northernmost and largest liquefied natural gas (LNG) facility worldwide is erected on the Melkøya Island in the northwestern part of Norway near Hammerfest [1]. The natural gas comes from the subsea fields Snøhvit, Albatross and Askeladd located in the Barents Sea about 150 km northwest from the LNG plant and are estimated to an amount of more than 193 billion cubic metres of natural gas. Due to the very low temperatures and some newly developed processes, such as the Mixed Fluid Cascade (MFC<sup>®</sup>) Process the energy efficiency of this plant is very good compared to similar ones of same size [2]. New ways have also been searched and found for the installation of the system which is done for nearly the whole plant on a barge which is then shipped to its final destination, see Fig. 1:

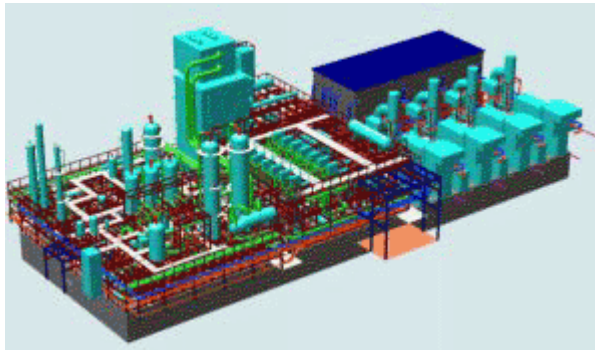


Fig. 1 The Snøhvit Barge [3]

Further improvements are achieved in the area of the engineering and commissioning of the plant. As the start-up of the plant is a very time consuming task and because some processes were newly introduced it was decided to do as much engineering work as possible before the real start-up of the plant. This is why Statoil and Linde decided to perform a virtual commissioning of the plant with the help of a high fidelity simulator of the plant with the goal to start training one year in advance of the real commissioning and to perform as much of the tuning of the parameters and the testing of the control solution as possible.

## 2 The Simulator

The simulator consists of a dynamic process model provided by Fantoft Process Technologies, Sandvika, (now a part of Kongsberg Maritime), the Human System Interface (HSI) of the 800xA system manufactured by ABB, an engineering simulator which emulates the original behavior of the PM875 components manufactured by ABB and an emulator of the HIMA security system.

While it is common practice to simulate the process this is not the case for the DCS. Earlier the DCS was simulated with libraries of the function blocks which were provided by the simulator manufacturers. This

had the great disadvantage that the functionality of the DCS was not simulated exactly but was to some extent inaccurate. It was also a time consuming task to follow up changes in the control logic. The emulator of the AC 870P / Melody DCS provides a one-to one copy of the original behavior and is changed very fast due to the fact that the original data - as they are loaded to the real controllers - is used to configure the emulator.

The HSI of the 800xA system is used as standard product with some simulator specific extensions and is stimulated by the simulator. This assures for the operators to be trained with a look and feel identical to the real system thus improving training effects.

### 2.1 Process Model

The dynamic model comprises [4]:

- For the onshore plant:
  - All parts of the processing plant, from multiphase pipeline slug catchers to LNG off-loading
  - Mixed Fluid Modified Cascade LNG Process
  - Unique multi-stream heat exchangers
- For the offshore plant
  - Wells, subsea equipment, flowlines and multi-phase pipelines to shore

As simulation tool D-SPICE (Dynamic Simulator for Process, Instrumentation and Control Engineering) is used for the process model and OLGA2000 for the pipeline model [5]. The configuration is done by means of a graphical Human Machine Interface:

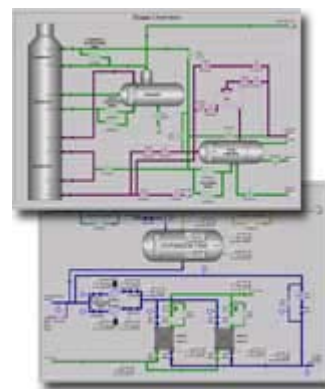


Fig. 2 D-SPICE HMI [5]

### 2.2 Digital Control System (DCS)

As DCS for the LNG plant the System 800xA is used [6][7][8]. This is a large-scale DCS for medium and large sized plants. Its active part AC 870P / Melody is a bus based system with decentralized controllers (PM875). The basic structure is shown in Fig. 3. The

data coming from and going to the process are transferred through I/O-Units or through MODBUS components. Via the high speed bus Fnet they are transferred to the Control Units. All communication between the Control Units is done via the control network Cnet. The data needed for display and operation in the HSI of the 800xA system are transferred via the Onet. These networks have been separated to be sure that the active part of the systems is not influenced in any circumstances by a large amount of data transfer to the HSI (as this may e.g. happen during some plant failure, where a large amount of alarms is transferred for display in the alarm lists).

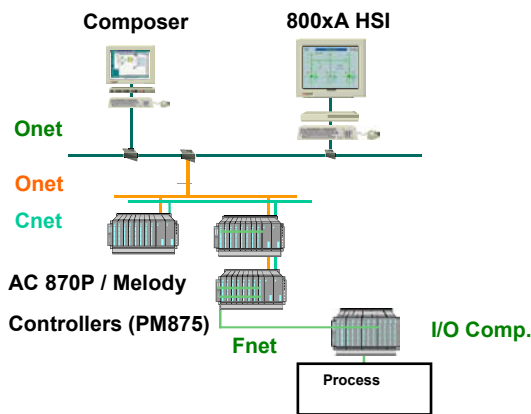


Fig. 3 800xA System Structure

There are 18 Control Units in use and more than 10 000 in-/output values connected to the process. Each Control Unit includes a Coupling Unit which manages the communication to the HSI. The controllers use the pSOS+ operating system and allow up to 16 different tasks with different cycle times and preemptive priority scheduling.

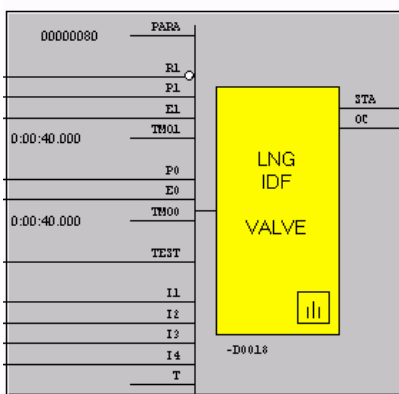


Fig. 4 Function Block

The configuration of the DCS is done in the engineering station "Composer" which is coupled to the Control Units via the Onet. Changes in the configuration of the system are only possible through Composer ("forced forward documentation"). The configuration is done graphically in the function plans.

An example of a user defined function block is shown in Fig. 4.

On the left side are the inputs, which are process signals and parameters. Parameters may be fixed, calculated from other process signals or changed by the operator via the HSI. On the right side are the outputs (process signals and status information).

As support for the commissioning and follow up of system malfunctions during operation it is possible to look inside the DCS and to measure and display internal and global variables.

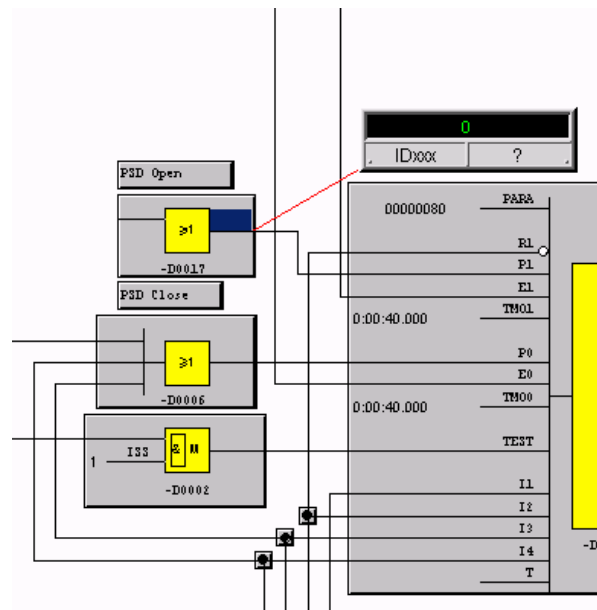


Fig. 5 Measured Output Value

### 2.3 Emulator of the DCS

For the simulator to be presented here, the HSI of the 800xA system is used which is delivered to the real plant, too. In order to provide simulator functionalities some add-ons are integrated, the so called ITS. They allow to e.g. save initial conditions and start again from a loaded one, to increase or decrease simulation speed etc.

The Control and Coupling Units themselves are substituted by an Emulator which runs on one DELL Poweredge 2950 server with 2 dual core 2.33 GHz CPUs with Windows 2003 Server operating system. The Emulator uses the original source code of the function blocks thus ensuring exactly the same behavior as the original system. The task scheduler of pSOS+ is implemented in a way that the different cycle times are recognized. The different priorities are not taken into account as they are only of some influence if the Control Units are overloaded. The network and the transfer delays which arise from the transport of the data via the network are not emulated. Each Control and Coupling Unit runs in a different Windows task which allows to easily expand the system.

All calculated data are stored in the data base Primo·S which is manufactured by ABB Automation Products GmbH. The data base is very flexible and allows different ways to allow data access thus enabling an extraordinary speed. The use of this data base allows to save initial conditions of the whole system which would not be possible with a stimulated real system.

The data which describe the control application as a whole are stored in different places:

- The configuration data sent from the Composer to the Control Units are stored in the so called Disassembler files. They store the structure of the application, i.e. the type of function block, which has to be calculated, the commissioning parameters of each single block (e.g. amplification or integration time of a controller), the input and output variables and the indices of the internal variables, which most of the blocks need to store the state of the last calculation step(s).
- The block internal variable, the local variables which are exchanged between function blocks on the same Control Unit and the global variables which are used for data exchange between different Control Units, with the process and the HSI are all stored in the data base and are part of the initial conditions.

To get an impression of the complexity of the system the following quantities should be mentioned:

- 94 000 function blocks
- 360 000 commissioning parameters
- 650 000 variables inside the Control Units
- 350 000 global variables for communication with the process, the HSI and between the Control Units.

The cycle time is 500 ms.

The connection to the process simulator is achieved with an OPC DA server provided by the emulator. It exposes all global variables to the clients. The update time is 500 ms as in the emulator. The HIMA security system uses a data transfer via MODBUS which in the emulator is changed to OPC.

During startup the OPC DA server scans through all the I/O data in the data base and through the signal lists provided by the Composer and creates the so called "extended MSR list" which contains all signals which have to be exchanged with the process simulator, including the exact name in the emulator data base and the measuring range. This allowed Fantoft to create scripts to automate the coupling.

The configuration of the emulator is done using exactly those output data of the Composer which are used to configure the original system (the so called "disassembler files"). No manual intervention is

necessary. So it is assured that the emulator shows exactly the same behavior as the real system.

In order to be able to check the proper functionality of the control logic Composer may be coupled to the emulator as it is coupled to the real system. So e.g. it is possible to measure internal or global variables and to display them in the function plan exactly in the same way as it is done with the real system. So the commissioning engineer does not have to learn special knowledge if he is working with the emulator.

### 3 Lifecycle Support

The emulator is designed to support the whole lifecycle process of the Snøhvit LNG plant. The general tasks to be performed are [9]:

- Concept
- Hardware design
- Engineering
- Start-up
- Operation
- Upgrade

Whereas concept and design are tasks which are very strongly dedicated to hardware design, the emulator is useful in the rest of the tasks. At a very early stage the control application is of some importance for the optimized and efficient operation of the plant and has to be checked for its abilities.

The main features to be checked during the engineering phase were

- Control system checkout: It has to be checked whether the basic functionality has been properly implemented and whether the control system is able to perform all the tasks needed. This starts from very simple things like "Are the actuators working in the proper direction?" to more demanding functionalities as e.g. are surge protection for the compressors (to ensure safe operation) or the correct function of the sequence controls (for safe and efficient operation of the plant).
- Procedure checkout: The start-up and operation procedures were provided by the process suppliers. As the plant is new and a lot of processes were introduced for the first time in order to minimize the risk it was necessary to check out whether the assumptions made were correct. Due to the process simulator which is assumed to at least represent the process to a good degree and because the original DCS behavior was available in the emulator the customer is confident that the results of this checkout will at least be good starting points for the commissioning of the real plant. More than 500

items where detected and resolved in this area [9].

- Upgrade checkout: In the future the simulator may be used to check in advance whether changes in the hardware or the control strategies will perform properly. They may be tested thoroughly without risk for the real plant and implemented if successful.

#### 4 Emulator Upgrade after Changes

For all the above mentioned tasks it is necessary to be able to upgrade the existing control application present in the emulator. In order to minimize the work it is vital to preserve all data already gained and to upgrade only those which have been changed.

The following upgrades are possible:

- Changing the structure of the control application. This happens if during the integration test a malfunction of the control application engineered so far has been found. Here we had to investigate how the real system worked and to find out how this behavior could be achieved in the DCS emulator.
- Changing of parameters of the function block. These parameters are real constants which are only to be changed via Composer.
- Changing of parameters which may be changed via the HSI, so called “operatable parameters”
- Identifying the valid parameters to be able to check whether everything has been changed properly.

Additionally it has to be taken into account, that the upgrade of the control application is done in the ABB engineering facilities in Bergen, but the integration of the DCS emulator with the process model is done at the Fantoft facilities in Sandvika near Oslo. Finally the training simulator has been already installed in Hammerfest by start of 2006 and it has to be upgraded to the latest version to be able to perform proper training sessions.

##### 4.1 Upgrading the Control Application

When the control application is changed function blocks are deleted, added or the in- and outputs of one or several function blocks are changed. This procedure is only done in Composer, as already mentioned above. Every function block has a unique identification (Betriebsmittelkennzeichen, BMZ). Further on most of the blocks need internal (local) variables to store the state of the last calculation step(s). If a function block is deleted, Composer keeps the corresponding BMZ and all local variables in a list, which prevents this BMZ or local variable ID to be reused immediately. This is only possible after having loaded the changed application to a Control Unit and to perform a change again.

If a function block is added, the BMZ and the local variables are added at the end of the up to then used indices. Gaps in this list of indices are only filled after having loaded the control application to a Control Unit.

These rules allow to implement a rather simple algorithm which compares the upper limits of the indices of each Control Unit of the new, changed control application with those of the old application and to set them to the default values (normally 0).

##### 4.2 Upgrading Constant Parameters

The constant parameters which are changed via the Composer are part of the disassembler files. So after having changed them they are valid from that moment on. They are normally changed during the virtual commissioning of the plant, if the default values or the values set by the engineering are not valid.

A problem comes up, if the disassembler file is replaced by a new one due to changes in the control application. Then these data are lost. This is why a tool was integrated in the emulator which allows to export all parameters of the system. A similar tool is available in Composer. Another tool allows to import these parameters again into the emulator. A similar tool is available with the Composer. These export/import tools allow to provide two different solutions for the problem:

1. The exported parameters from the emulator are imported in the Composer before the new disassembler files are generated. Then the new parameters are already included in the new application because the disassembler files will be generated including the new parameters.
2. The exported parameters are imported to the emulator after loading the changed disassembler files. Then the already loaded disassembler files will be changed and the parameters are available in the emulator, too.

Both solutions are necessary during the ongoing project. In the early stage version 2 might be normal, because the engineering people try to improve the control application as fast as possible and don't care too much about parameter changes. But in the final stage of the project version 1 is to be chosen as it is necessary to import the parameters into the Composer because the real Control Units are configured by the Composer and there is no other way to get the parameters in there.

##### 4.3 Upgrading Operatable Parameters

The operatable parameters are those which might be changed by the operator via the HSI. It is a question of the configuration of the control application which values may be changed. Normally these parameters are the high/low limits of a monitoring which then creates alarms if violated. Values for the operatable parameters may be defined in the Composer and are

then included in the disassembler files. Nevertheless they may be changed by the operators at any point of time.

These parameters are global variables (because they come from the HSI) and therefore they are part of the initial conditions. This is why after loading an initial condition the parameters stored in the disassembler file are overwritten. So if one parameter is changed it is only saved in initial conditions which are saved after that change. That is why these parameters have to be exported and re-imported again if they shall be preserved after a change. We have implemented a tool which imports these parameters to all currently existing initial conditions.

In the final stage of the project here, too, it is recommended to import the operatable parameters into the Composer. Then these parameters will be loaded to the real Control Units as the starting values.

#### 4.4 Upgrading Initial Conditions

Finally the upgrading of the initial conditions shall be described. The upgrade of initial conditions is of vital interest, because the whole integration work already done is saved in there.

This upgrade consists of various steps to be performed. First the memory allocation has to be checked for added or deleted internal variables. They have to be set to the default value (normally 0). All other internal variables and the global variables are copied one to one. Then the parameters have to be upgraded. They first have to be exported from the old application and then to be imported to the new one. The different procedures for the constant parameters and the operatable parameters are performed automatically.

### 5 Upgrade Workflow

The integration and testing process takes place in two different places: the engineering facilities in Bergen and the simulator facilities in Sandvika. Training is performed in Hammerfest.

While the integration group is performing the integration work (adapting parameters to the plant behavior, e.g. controller parameters, waiting times of sequence controls, high and low values of monitoring functions ...) they look for malfunctions in the control logic and report them to the engineering group. Integration continues while the malfunctions are changed in the control application by the engineering group. Additionally new functions are engineered and integrated in the control application.

When a new version is released, the parameters and initial conditions of the up to that point of time valid version are exported from the DCS emulator. A new version of the control application is generated and used to configure the DSC emulator.

The parameters are filtered with some scripts extracting the relevant data. With a total amount of parameters of more than 360 000 this helps to keep track of the changes. The relevant data are the controller parameters and the open/close times of the actuators. Then the emulator is started and the parameters are re-imported. The tools provided assure that the data are saved in the disassembler files and in the initial conditions.

This loop has to be performed for every new release of the control application. Currently version 27 is in use.

Before commissioning the real plant the extracted parameters have to be re-imported to the Composer. Then the control application together with the parameters found during the virtual commissioning are loaded to the real Control Units.

The upgrade of the training simulator is normally done as a one-to-one copy of the latest version of the integration simulator. This is easily achieved by creating images of the hard disks.

### 6 Planned Further Extensions

As the plant is very large with a lot of auxiliaries it is nearly impossible to perform training sessions with the whole operator staff at the same time. At the simulator side the whole process model has already been divided into 10 process sections which may run independently from each other. For the emulator the same task has already been investigated and it is planned to implement the section handling in the near future. It will then e.g. be possible to perform a training session for the sub-sea section operators only (or any other section or any other combination of sections).

The tasks to be solved here are to identify the signals going to and coming from the section and to supply them with valid signals. This task has to be done in close cooperation with the process simulator supplier who will supply these data.

### 7 Conclusions

The procedure how to implement lifecycle system capabilities in simulators and DCS emulators strongly depends on the way these emulators were implemented. It is always necessary to analyze thoroughly the internal data structure to be able to decide which measures are to be taken. Standard recipes are not on hand.

### 8 References

- [1] <http://www.statoil.com/statoilcom/snohvit/svg02699.nsf?OpenDatabase&lang=en>
- [2] Berger, E. et al.: *Das Snøhvit-Projekt*. Linde Technology, 1:12–23, 2003. (<http://www.linde.com/international/web/linde/like35linded.nsf/rep>)

ositorybyalias/pdf\_lindetechnology\_1\_2003/\$file/  
Linde\_Technology\_1\_2003\_DE.pdf)

- [3] <http://www.offshore-technology.com/projects/snohvit/snohvit7.html>
- [4] <http://www.fantoft-process-technologies.com/downloads/Simulators%20Snohvit.pdf>
- [5] <http://www.km.kongsberg.com/ks/WEB/NOKBG0240.nsf/AllWeb/5552FF4BD0255B25C1256A68002B4CAB?OpenDocument>
- [6] Krause, H.: *Emulator for the digital control system „Symphony“*. ESS'2001, Simulation in Industry, 13th European Simulation Symposium and Exhibition, Marseilles (France), October 18<sup>th</sup>-20<sup>th</sup> 2001, SCS-Europe Ghent
- [7] Krause, H.; Niss, T.: *Emulator für das Leitsystem „Melody“ der ABB*. VDI-VDE-Aussprachetag „Rechnergestützter Entwurf von Regelungssystemen“, September 13<sup>th</sup>/14<sup>th</sup> 2001, Dresden
- [8] Krause, H.: *Emulator für das Prozessleitsystem „Symphony Melody“ der Firma ABB*. Paper presented during GMA-Congress 2003, June 3<sup>rd</sup>/4<sup>th</sup> 2003, Baden-Baden
- [9] Skjerven, Ø., Vist, S.: *Snøhvit Lifecycle Simulator from Wellhead through Pipeline and LNG Liquefaction to Offloading*. 15th International Conference & Exhibition on Liquefied Natural Gas (LNG15), April 24<sup>th</sup> - 27<sup>th</sup> 2007, Barcelona.