PLC Simatic S7

Programming

SIEMENS

English Version 1.4

Content

1. Preface

The BatchXpert system is a system for controlling and visualizing processes throughout the plant, incorporating batch reporting, protocols and recipes, batch tracking with materials, trends, alarms, BatchXpert station and/or HMI, etc.
For simpler applications there is the BatchXpert Compact system, this system is for processes that do not need protocol reporting. batches or recipes, but incorporates trends and a history of alarms, a
HMI, etc...

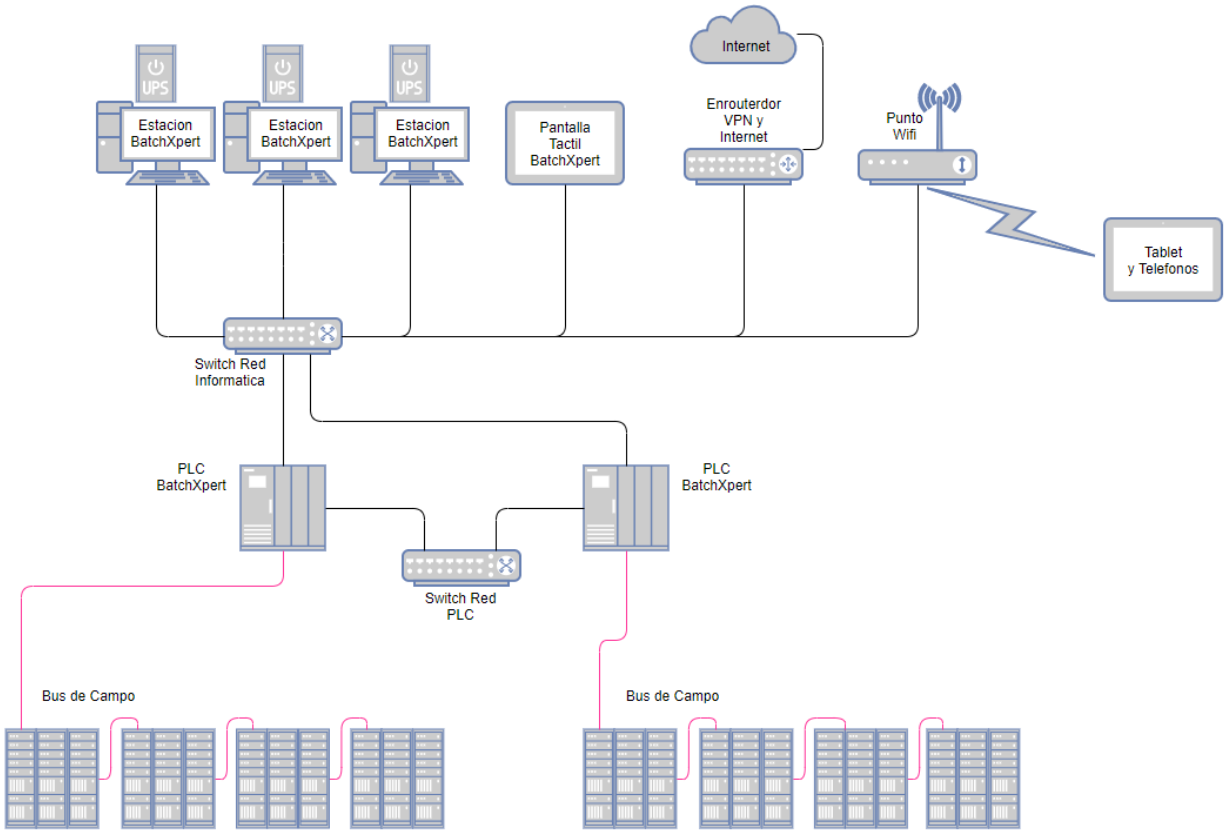
The PLC program was made as a program on the basis of S7, so it can be used on all S7 PLCs. For the BatchXpert, a special standard was implemented in programming, with modularization, standardization of programming with the creation of standardized interfaces, to achieve a considerable improvement in process engineering times. In this Program, several Criteria were considered – from short cycle times to the preparation of the program with various prepared and standardized processes.

Some of the features of the BatchXpert system standardization:

- 120 Units (Sequences)
- 1600 Actuators (Motors, Valves,...)
- 1600 Digital Inputs (Initiators, Vacuum Sensors,...)
- 480 analog inputs (temperatures, pressures,..)
- 320 Meters (Flow Measurements)
- 320 Regulators (PIDs) ; and several other Modules.

2. Global Communication View

The BatchXpert system uses standard communication interfaces and supports various communication systems from the PLC to the Automation Field. The following is a summary of the communications:



The BatchXpert system supports connecting "BatchXpert Stations", "HMI Displays" and other equipment through an Ethernet network with TCP/IP, which can be managed with standard tools for Ethernet network management. It is also possible to connect laptops through "WiFi" access points and provide a form of remote control for smartphones or tablets.

Communication between PLCs takes place at a separate PLC gate, thus preventing BatchXpert network communications from interfering with the exchange of data between system controllers.

The customer has the option of connecting the system to their corporate network and/or the internet to access the databases and to enable the possibility of Teleservice (through "TeamViewer", please read the "TeamViewer Teleservice Manual").

The communication of the Cambo is carried out through "Profibus" or "ProfiNet" interfaces that are industrial standards and allow to connect a wide variety of equipment such as IO systems, olive harvesters, meters, etc.

3. Overview of HMIs

3.1.Screen Resolutions:

The BatchXpert system supports the following HMI systems with the following resolutions:

- VisXpert **Visualization**
  - 1600 X 900 (HD)
  - 1920 x 1080 (Full HD)
  - 1366 x 768 (Notebook)

- Movicon **Display** for Touch Screens
  - 1024 x 768" (10")
  - 800 x 480 (7")

- **Flexible WinCC Display** for Touch Screens (without TIA Portal)
  - 800 x 480

- WinCC Basic and WinCC Comfort **Display** for Touch Screens (with TIA Portal)
  - 800 x 480

3.2.Adjust Screen Resolution

VisXpert

- There is a GraphPic library in "BatchXpert SDK"/ Visu/ GraphPic Vx.x.x.x (Unzip in project folder)
- To select the resolution of the project in the GraphPic we must go to the "GrapHPic Editor" and in the list of windows we will find "BaseProcessWindow" (select the corresponding resolution).

Movicon

- There is a Movicon library in "BatchXpert SDK"/ Visu/ Movicon Vx.x.x.x (Unzip in the project folder).
- Select the folder with the resolution you need, example: for a 7" screen we keep the folder "Dev 800x480" (this folder should be renamed with the name of the project)
- Delete the remaining folders (the folders we don't use from the library)

Note: All resolutions are subject to change. However, this usually requires several considerations, so we recommend consulting Mlogics for such a problem.

3.3.Hostnames

In a BatchXpert system, the default operating stations are set to the following names, which are adjustable, but it is recommended to maintain the system standard.

- BX01 for the first server
- BX02 for the second server
- BX03 for the third server
- BXSlave1 for the First Slave
- ...

The program to take a station name follows the following order:

1. tags. Ini: Looks for the name found in: "**C:/Daten/BX.ini"** in the **"StationName = " setting**
2. HostName: If there is no name in "StationName" it assumes the HostName (name of the PC)
3. Slave: if it doesn't have a name, it assigns a "0" at the end of the name

Note: We as programmers must have our BatchXpert station **as a slave**, this is so as not to generate conflicts with some "master" station that are in the industries.

To assign our computer as a slave go to "BatchXpert Management Console" in "BatchXpert Station" and change to "Slave".

3.4.Important Directories

BatchXpert relies on a few important folders in order to function properly. In:

C:\Daten

There are several folders that correspond to different system functions.

- **Backup**: Contains system backups
- **Engineering:** contains engineering details such as taglist imports
- **Logs**: contains information about the events of each tool

C:\Program Files (x86)\BatchXpert

there are folders where we can find information and also the executables of the BatchXpert tools.

- **Documentation:** contains manuals for programmers and operators in different languages on how to use BatchXpert, as well as information on the BatchXpert license.
- **Reports:** in this folder you can find the templates of the different reports that can be generated by BatchXpert (Batch Sumy, Recipe Report, Report With Menu...).
- **Tools: These** are additional tools from the BatchXpert (Comandline Tools).

C:\Program Files (x86)\BatchXpert SDK

There are additional BatchXpert tools and templates for the programmer, it is oriented to the compatibility of the different softwares with the BatchXpert system and HMI, to minimize the engineering time, the **installer of the BatchXpert SDK** is required.

- **Documentation:** contains manuals for programmers on how to use the BatchXpert, HMI and PLC tools.
- **PLC:** contains templates for the S7 PLC (Vipa compatible)
- **Tools: Contains** templates for project documentation (tagliste, Project Information) and tool executables (Installation Center, Comandline Tools).
- **Visu:** you will find the templates for the visualizations (GraphPic, Movicon, WinCC Flexible).

C:\Program Files (x86)\BatchXpert Micro SDK

There are templates for the programmer, it is geared for reduced PLCs in memory, HMI compatibility and to minimize engineering time, the **BatchXpert Micro SDK installer is required.**

- **Documentation: :** contains manuals for programmers on the use of the HMI and requirements of the BatchXpert Micro.
- **PLC:** Contains templates for the S7-1200, S7-300 and RSLogix5000 PLC.
- **Visu:** you will find the templates for the visualizations (Movicon, Factory Talk View).

4. PLC Overview

The BatchXpert system consists of a basic program, or "Operating System" in a certain form, that abstracts the user's program from direct access and management of the periphery, and provides additional advanced functions such as Timers, Status, Control Modules,...

The PLC Program is generally structured as follows:

- FC 1-100: Fixed System Functions (Block Numbers Cannot Be Reassigned
- DB1-100: Fixed system data, with no possibility of reassigning.

The rest of the functions and FB not mentioned, are free for the use of the user (programmer). However, there are many auxiliary functions that generally occupy the FC 400-600 range, but which can be redirected by the user, if necessary.

The fixed FC system of the BatchXpert system may seem very rigid to a programmer, but in reality this convention facilitates the programming of both the system and the user's project.

4.1.Requirements to the PLC

The BatchXpert system requires certain features of the PLC. The system mostly requires a lot of RAM to be able to funionize. For more information, please refer to the system's "System Requirements" manual.

The program is compatible with the Siemens **S7-300, S7-400, Vipa Speed7 and S7-1500** lines. The S7-1200 series PLCs are compatible with the smaller BatchXpert Micro system, but can use the same HMI.
Programming is carried out with the software "Simatic Manager" or "Tia Portal V16+" as appropriate for the PLC used.

4.2.Program Structure

The following shows the structure of the system's general calls.

The functions are colored according to the following categories:

- System functions, not modifiable
- IO-related functions, Adjustable if required
- FC5 (Bx WinOrder)

OB1 (CYCL_EXC)			
	FC10 (Bx SysTime)		
	FC1 (Bx SysBegin)		
		FC8 (Bx SysInit)	
		FC86 (Bx UnitProtSend)	
		FC50 (Bx RecLoader)	
		FC96 (Bx UnitPc)	
		FC94 (Bx UnitProgWin)	
		FC97 (Bx UnitStatusInfoWin)	
		FC7 (Bx ManuProtSend)	
		FC45 (Bx DiagDP)	
		FC502 (TransDIn)	
		FC16 (Bx DIn)	
		FC503 (TransAIn)	
		FC21 (Bx AIn)	
	FB101 (U001 config)		
		FC100 (Bx Unit)	
			FC101 (U001 Phases)
	FB110 (U010 config)		
		FC100 (Bx Unit)	
			FC102 (U002 Phases)
	FC2 (Bx SysEnd)		
		FC11 (Bx Act)	
		FC31 (Bx PID)	
		FC36 (Bx Msg)	
		FC39 (Bx Switch)	
		FC4 (Bx SVal)	
		FC501 (TransAct)	
		FC5 (Bx WinOrder)	

Generally, user programs are programmed into the FB1xx and FC1xx of the sequences. IO-related blocks are generated by the Engineering tool of the BatchXpert system.

4.3.Important Global Signals

The BatchXpert system provides the following signals for use in the user's program. All signals presented are "READ ONLY" and should not be written by the user. The following Signals are global, and can be used throughout the program.

Number of PLCs within the BatchXpert System

PLCRestart	M 878.0	BOOL	PLC restart (Stop => Run)
PLCRunning	M 878.1	BOOL	PLC Running after restart
ToDo_Read	M 878.2	BOOL	To Do - read signal
ToDo_Write	M 878.3	BOOL	To Do - set signal
QuittAll	M 878.7	BOOL	reset all alarms
PLCNo	MW 998	INT	Numero de PLC dentro del Sistema BatchXpert

time milli seconds in DINT

Clk2CE	M 879.0	BOOL	clock 2 cycle (edge)
Clk4CE	M 879.1	BOOL	clock 4 cycle (edge)
Clk8CE	M 879.2	BOOL	clock 8 cycle (edge)
Clk16CE	M 879.3	BOOL	clock 16 cycle (edge)
Clk32CE	M 879.4	BOOL	clock 32 cycle (edge)
Clk64CE	M 879.5	BOOL	clock 64 cycle (edge)
Clk128CE	M 879.6	BOOL	clock 128 cycle (edge)
Clk256CE	M 879.7	BOOL	clock 256 cycle (edge)
Clk01	M 880.0	BOOL	clock 0,1 sec (10 Hz)
Clk02	M 880.1	BOOL	clock 0,2 sec (5 Hz)
Clk04	M 880.2	BOOL	clock 0,4 sec (2,5 Hz)
Clk05	M 880.3	BOOL	clock 0,5 sec (2 Hz)
Clk08	M 880.4	BOOL	clock 0,8 sec (1,25 Hz)
Clk10	M 880.5	BOOL	clock 1,0 sec (1 Hz)
Clk16	M 880.6	BOOL	clock 1,6 sec (0,625 Hz)
Clk20	M 880.7	BOOL	clock 2 sec (0,5 Hz)
Clk1E	M 881.0	BOOL	1 second (edge)
Clk1E1	M 881.1	BOOL	1 second (edge) , 1 cycle later
Clk1E2	M 881.2	BOOL	1 second (edge), 2 cycle later
Clk6E	M 881.3	BOOL	6 second (edge)
Clk10E	M 881.4	BOOL	10 second (edge)
Clk60E	M 881.5	BOOL	60 second (=0.1 minute, edge)
Clk1DayE	M 881.6	BOOL	1 day (edge)
1sec	M 895.0	BOOL	tact 1 sec
2sec	M 895.1	BOOL	tact 2 sec
4sec	M 895.2	BOOL	tact 4 sec
8sec	M 895.3	BOOL	tact 8 sec
16sec	M 895.4	BOOL	tact 16 sec
32sec	M 895.5	BOOL	tact 32 sec
64sec	OB1 (CYCL_EXC)	BOOL	FC10 (Bx SysTime)

128sec	FC1 (Bx SysBegin)	BOOL	tact 68 sec
FC8 (Bx SysInit)	MB 879	BYTE	FC86 (Bx UnitProtSend)
CycleTimeMin	MD 904	FC50 (Bx RecLoader)	time minutes
CycleTimeHour	FC96 (Bx UnitPc)	REAL	time hours
FC94 (Bx UnitProgWin)	MD 912	REAL	FC97 (Bx UnitStatusInfoWin)
CycleTimeSec	MD 900	FC7 (Bx ManuProtSend)	Cycle Time in Seconds
TimeMSecInt	FC45 (Bx DiagDP)	DINT	time milli seconds in DINT

4.4.Program Creation with the Help of the HMI

Assuming that the controller programming is done after the HMI charts are created, the chart can be used as an aid. With a click on the object the number is visible, which is also used in programming, simple numbers are easier to maintain in the short term than a complex code tags.

There are multiple units for a class, so it's a good idea to generate a DB of classes.

1. All objects are classified in DBs
2. Programming from phases, Evaluation, Blocks, ... About DB Classes
3. Objects are classified in the new DB of classes

5. Difference Between BatchXpert and BatchXpert Compact

The two systems, BatchXpert and BatchXpert compact, are very similar, and for the most part the user code can be maintained without any changes in the PLC. However, the BatchXpert Compact system is designed for smaller projects without the need for reporting.
The "BatchXpert Compact" system incorporates the control modules of the "BatchXpert" system and has the same functions and control standards. The system is perfect for controlling processes and machinery that do not require advanced historical data recording systems and process management functions as there are in the "BatchXpert" system. The "BatchXpert Compact" system includes basic systems for recording historical data such as trends and alarm logs.

There is no limit on the number of HMIs.

	BATCHXPERT	BATCHXPERT COMPACT
FC16 (Bx DIn)	Microsoft SQL2005 o mas nuevo.	No hay base de datos. Todos los datos están en el PLC.
FC503 (TransAIn)	Tendencias, Alarmas, Registro de procesos, Registro de operaciones manuales, Reportes de procesos.	Tendencias y Alarmas.
FC21 (Bx AIn)	Hasta 8 servidores completamente independientes.	FB101 (U001 config)
Manejo de secuencias	Manejo avanzado con programas guardados en la base de datos. Editable con editor de configuración del sistema.	FC100 (Bx Unit)
Conexión a PLC´s simultáneos	Ilimitada.	Un PLC por cada Pantalla táctil.
FC101 (U001 Phases)	Todos.	FB110 (U010 config)
Recetas y programas	Ilimitados y guardados en la base de datos.	FC100 (Bx Unit)
Límite de HMI	No hay límite (se permite hasta 8 servidores, pero un número ilimitado de clientes).	No hay límite en cantidad de HMI.

6. BatchXpert SDK and BatchXpert Engineering Tool

To facilitate the generation and execution of an automated project with BatchXpert, there is the "BatchXpert Software development Kit" also called "SDK". This package installs all the engineering tools and templates for both the controllers and the display systems.

The most recent version of the SDK can be obtained from the following link:
http://www.mlogics-automation.com/?q=es/content/descargas-para-clientes

It is recommended that you use the same version of the SDK as the version of BatchXpert that is used in your plant. For new projects, it is recommended to use the latest versions, to take advantage of system improvements.

Once installed, you will find the following folders inside the installation folder:

- **Visu**: This folder contains the project templates for all supported visualization systems of the BatchXpert system. The appropriate file should be copied to the engineering folder, extracted, and adjusted to suit the needs of the project.
- **PLC**: Contains the templates of the supported PLCs of the BatchXpert system. As with the visu, you need to copy it to the engineering folder, extract it, and adjust it.
- **Tools**: Contains useful engineering tools, such as Taglist Templates, a SQL Database Management Tool, and several other tools
- **Documentation**: Contains many Manuals, which are not included in the normal installation of the BatchXpert, as they are intended for an Engineer and not for Operators.

Several tools will also be installed, such as the "BatchXpert Project Engineering Tool" that allows importing, exporting and generating data for PLCs and HMIs based on the system's current engineering database. This tool can generate alarm messages for the HMIs, data blocks for the PLC and others.

7. General Structure of Control Module DBs

In the PLC, the data of the object (actuator, PID, ...) is kept in Arrays, you can also use DBs with the division of different elements. It is only important that the internal structure of the objects is maintained. Of course, the use of the different elements has the advantage that on each element you can write the name of the tag and a comment. However, this only makes sense if you can use a tag code throughout the project. If the tag code is specified by the client, then the tag names of the elements are known, usually only if the project is almost complete.
The division into Arrangements is preferred. When creating a project, it can be done as follows:

1. Enter the number of objects in the P&ID - Diagram
2. Creating Graphic Images
3. Reading Excel Tables with the Object Name
4. Programming the S7

PLC templates all maintain the same structure for programming, whether it's a BatchXpert SDK template or Micro SDK (the Micro SDK has no sequence, units, etc.)

7.1.Data Structure

The data structure is directly related to the DBs and there is a block of data for each control element (actuators, PIDs, etc.). These Data Blocks contain a long array, where each item corresponds to the corresponding number module and follows the structure described in this section.

7.2.Commands

This section describes the data that functions as commands from the user's program to the BatchXpert system. The signals described in this section can be written into the user's program with the corresponding restrictions for each signal.
These signals are usually commands that activate the corresponding function in the control modules. Generally (with a few exceptions), these are "Write only" signs.

7.3.Status

These are data and signals that the BatchXpert system provides for use in the user's program.
This is Status, where the user can obtain different information about the general status of the control modules. Generally (with a few exceptions), these are "Read only" signs.

7.4.Parameters

These are configurations of the different conrol modules. Normally this configuration is done through the "Faceplate" of the BatchXpert HMI systems. Normally, they are not written from the PLC.
However, for some signals there is the possibility of manipulating the parameters from the PLC to enforce a charactaristic record, without giving the user the possibility to manipulate the parameters. For example, reset simulations at emergency stop,...

8. Actuators (Act)

Actuators are all the outputs of the PLC that are or are not related to the process (valves, pumps, even lamps or LEDs).

An actuator basically has one output and up to two FeedBacks.

The actuator associated with the physical output is realized in the FC 501 "TransAct". (Example in "Assignment Act").

A PID in the display is presented as an actuator, to indicate the status of the PID actuator, however it is only a software actuator, usually it has no physical output.

8.1.duty timer value (seconds)

FC102 (U002 Phases)	Tipo	FC2 (Bx SysEnd)
ACo	BOOL	automatic control
FC11 (Bx Act)	BOOL	extern control
FC31 (Bx PID)	BOOL	status check start
FC36 (Bx Msg)	BOOL	feedback 1
FC39 (Bx Switch)	BOOL	feedback 2
FC4 (Bx SVal)	BOOL	Release
FC501 (TransAct)	BOOL	release 2
FC5 (Bx WinOrder)	BOOL	extern automatic
ACoHM	BOOL	automatic control help memory
ExCoHM	PLCRestart	M 878.0
BOOL	PLC restart (Stop => Run)	PLCRunning
M 878.1	BOOL	PLC Running after restart
ToDo_Read	M 878.2	BOOL
To Do - read signal	ToDo_Write	M 878.3
BOOL	To Do - set signal	QuittAll
M 878.7	BOOL	Reset All Alarms
PLCNo	998 MW	INT
Number of PLCs within the BatchXpert System	BOOL	Clk2CE
M 879.0	BOOL	clock 2 cycle (edge)
CLK4CE	M 879.1	BOOL
clock 4 cycle (edge)	CLK8CE	M 879.2
BOOL	clock 8 cycle (edge)	CLK16CE
M 879.3	BOOL	clock 16 cycle (edge)
CLK32CE	M 879.4	BOOL
clock 32 cycle (edge)	CLK64CE	M 879.5
BOOL	clock 64 cycle (edge)	CLK128CE
M 879.6	BOOL	clock 128 cycle (edge)
CLK256CE	M 879.7	BOOL
clock 256 cycle (edge)	CLK01	M 880.0
BOOL	clock 0.1 sec (10 Hz)	CLK02
M 880.1	BOOL	Clock 0.2 sec (5 Hz)

CLK04	M 880.2	BOOL
clock 0.4 sec (2.5 Hz)	CLK05	M 880.3
BOOL	clock 0.5 sec (2 Hz)	CLK08
M 880.4	BOOL	clock 0.8 sec (1.25 Hz)
CLK10	M 880.5	BOOL
clock 1.0 sec (1 Hz)	CLK16	M 880.6
BOOL	clock 1.6 sec (0.625 Hz)	CLK20
M 880.7	BOOL	clock 2 sec (0.5 Hz)
Clk1E	M 881.0	BOOL
1 second (edge)	Clk1E1	M 881.1

8.2.Free for Programmer-Specific Applications

BOOL	1 second (edge) , 1 cycle later	Clk1E2
M 881.2	BOOL	1 second (edge), 2 cycle later Sólo Activa en automático
CLK6E	M 881.3	BOOL Eficaz en automático y manual Enclavamientos relacionados con la producción están puenteados
6 second (edge)	CLK10E	M 881.4 Si el actuador no está apagado o alterado como una comprobación del estado se señaliza error
BOOL	10 second (edge)	CLK60E Es la realimentacion de la posicion del actuador
M 881.5	BOOL	60 second (=0.1 minute, edge) Es la realimentacion de la posicion del actuador
Clk1DayE	M 881.6	BOOL Son cosas vitales de seguridad como paradas de emergencia, puerta hombre.
1 day (edge)	1sec	M 895.0 Bloqueos de procesos no criticos pero importantes como nivel vacio, nivel alto.
BOOL	tact 1 sec	2sec
M 895.1	BOOL	TACT 2 sec

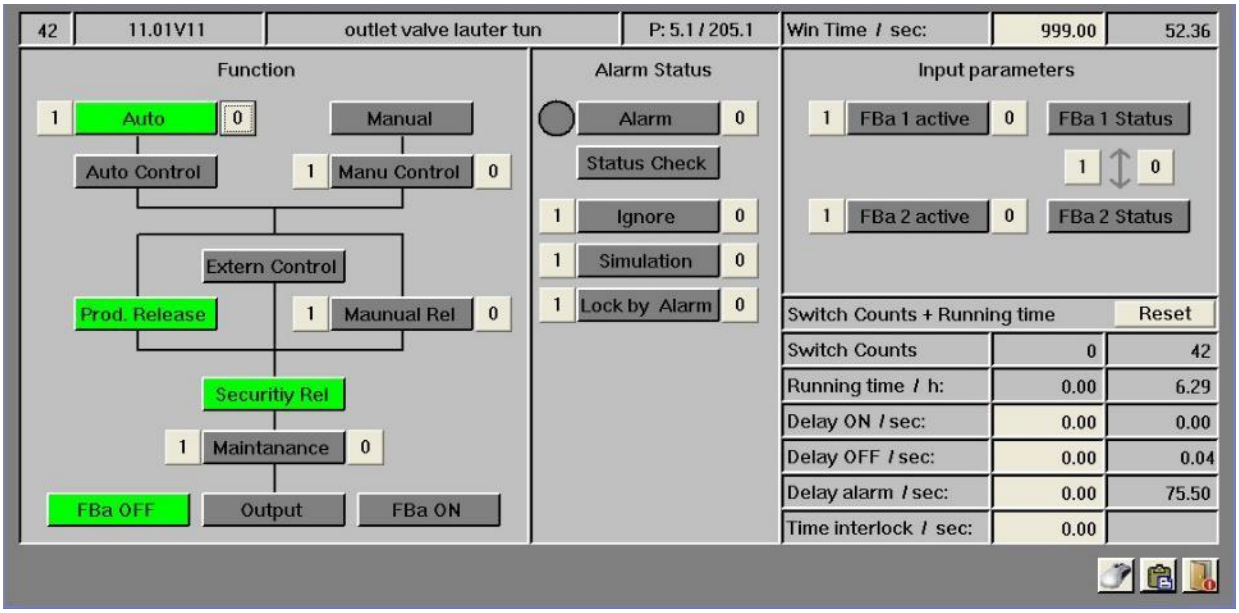
8.3.Exit

4sec	M 895.2
BOOL	tact 4 sec Este feedback puede estar activado o en simulación, esta señal esta generada Internamete.
8sec	M 895.3 Este feedback puede estar desactivado o en simulación, esta señal esta generada Internamete.
BOOL	tact 8 sec Estados de los actuadores no Ok
16sec	M 895.4 Alarma General fue o está activada Señal debe restablecerse por el operador (OK + Reset alarma)
BOOL	TACT 16 sec

	Estado incorrecto de actuadores para el próximo inicio del programa
32sec	M 895.5 Feedback aun incorrecto.
BOOL	TACT 32 sec
64sec	M 895.6
BOOL	TACT 64 SEC

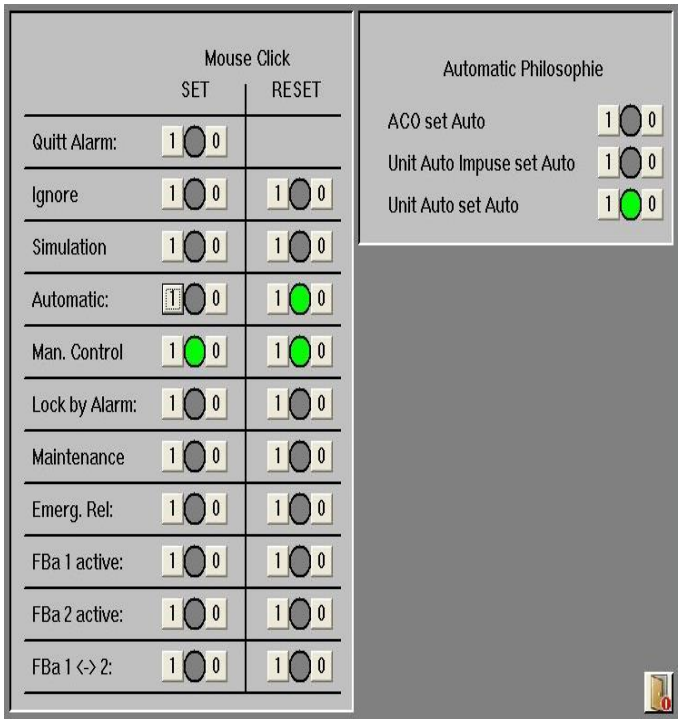
8.4. Timeout for reconnection

128sec	M 895.7
BOOL	TACT 68 sec 0 FBA1 = OFF = ON FBA2 1 = ON FBA2 FBA1 = OFF)
CycleCnt	MB 879
BYTE	cycle counter
CycleTimeMin	MD 904
REAL	Time minutes Si se envía un mensaje de error o no
CycleTimeHour	MD 908 El feedback se genera internamente, el Estado es siempre OK
REAL	Time Hours 1 = Automático 0 = Manual
CycleTimeDay	MD 912 Sólo es efectivo en el modo manual
REAL	Time Days Puentea una señal de un bloqueo relacionados con la producción
CycleTimeSec	MD 900 El actuador tiene alarma(GAL) por lo que no puede ser accionado.
REAL	Cycle Time in Seconds El actuador no puede ser actvado en manual o en automático No hay evaluacion de fallas
TimeMSecInt	MD 916 Un control automático se retrasa por segundo xxx
DINT	time milli seconds in DINT Un control automático se extiende al segundo xxx
ADSp	Retardo de supervisión de fallas Se inicia Cada tiempo de conmutación. Si el tiempo expira, la monitorización de fallas se habilitara.
TInterlock	Tiempo de espera para la reconexión El actuador se retarda hasta al menos xxx segundos antes de que una nueva Activación tenga efecto.



8.5.Special Configurations

In addition to the system window of the actuators, the default parameter settings are made, there is a window for the mouse parameterization. This determines what should happen when you click your mouse over the item. In addition, in the mouse parameterization you can even set the Manual/automatic behavior generally:



- Automatic control by actuator. If you drive an actuator, it's usually in automatic mode. Switching to manual mode is not always possible and when the actuator has a program effect. This corresponds to the automatic philosophy of many programs in the fermentation cellar (e.g. GEA).
- Automatic Edge Unit (RUN) is the only actuator mode in Automatic. Disabled, the RUN flank can be activated manually at any time.
- Auto Unit (RUN) sets the automatic mode of the actuator. While the corresponding unit is on RUN it cannot be switched into manual actuator mode.
- If one of these options is selected, the operator can manually interrupt it at any time.
- Switching from manual mode to automatic mode is always possible at any time.

8.6. Programming Examples

Automatic Process Control

U "PH"	activated while in that step and in "Start"
S "Act". Act[42]. Aco	Actuator 42 will be activated automatically
S "Act". Act[44]. Aco	Actuator 44 will be activated automatically

Signal Enabling Auto Mode

U "RUN"	Process in "Start"
S "Act". Act[42].xAuto	Enable Actuator Automatic Mode 42

```
S "Act". Act[44].xAuto          Enable Actuator Auto Mode 44
```

External Control

```
U "Din". Deen[15]. Gis          Safety switch
S "Act". Act[42]. ExCo          Activates the actuator from an external control
```

Release

Safety Release

```
U "Din". Deen[11]. Gis          Hmobre Gate
U "Din". Deen[10]. Gis          Emergency Stop
= "Act". Act[42]. Rel       Conditional Safety Release
```

Release by process

```
U "Act". Act[40]. Off       Valve 1 off
U "Act". Act[41]. Off       Valve 2 off
= "Act". Act[42]. Rel2          Process-Conditioned Release
```

Alarm Assessment

```
U "Act". Act[42]. GAlS"         Actuator on alarm
S "HoldReq"                 Maintains unity
```

Assignment Act

Since

```
U "Act". Act[1]. Out  Signal to be activate the physical output
= A 0.0                         Physical Output
A E 0.0                         Actuator Feedback 1
= "Act". Act[1].xFBa1           Turning on Feedback 1
UN E 200.0                  2 Actuator Feedback
= "Act". Act[1].xFBa2           Enable Feedback 2
```

Until

```
U "Act". Act[1600]. Out         Signal to be activate the physical output
= A 199.7                   Physical Output
UN E 199.7                      1600 Actuator Feedback 1
= "Act". Act[1].xFBa1
UN E 399.7                  Actuator 1600 feedback 2
= "Act". Act[1].xFBa2
```

9. Digital Inputs (DIn)

All DIn is considered as simple digital inputs, it is not considered as a feedback of actuators, door men or vacuum signals.

The digital input associated with the physical ticket is done on the FC 502 "TransDIn". (Example in "DIn Assignment").

9.1.Switch Counter Value

Asignación	Tipo	Comentario
EA0	BOOL	enable alarm by 0-signal
EA1	BOOL	enable alarm by 1-signal
BATCHXPERT	BATCHXPERT COMPACT	Database
Microsoft SQL2005 or newer.	There is no database. All the data is in the PLC.	Historical data
Trends, Alarms, Process Log, Manual Operations Log, Process Reports.	Trends and Alarms.	Redundancy of historical data
Up to 8 completely independent servers.	There are no historical data records except trends and alarms.	Sequence Handling
Advanced handling with programs saved in the database. Editable with System Configuration Editor.	Basic operation with programs stored in the PLC.	Connection to simultaneous PLCs
Unlimited.	One PLC for each touch screen.	Control Modules
All.	All.	Recipes & Programs
Unlimited and saved in the database.	Stored in the PLC and restricted to the PLC's memory.	HMI Limit
There is no limit (up to 8 servers are allowed, but an unlimited number of clients).	There is no limit on the number of HMIs.	signal extern help memory
B19	BOOL	spare
B20	BOOL	spare
B21	BOOL	spare
B22	BOOL	spare
B23	BOOL	spare
GAlQuitt	BOOL	general alarm quitt
Ign	BOOL	ignore alarm
Sim	BOOL	simulation
iEA0	BOOL	intern alarm by 0
iEA1	BOOL	intern alarm by 1
ImpProt	BOOL	write impule flank to protocol
ImpNegProt	BOOL	write negative impule flank to protocol
Switch	BOOL	convert as switch output
GAl	BOOL	Assignment
Guy	Comment	Aco
BOOL	automatic control	ExCo
BOOL	Extern Control	SCS
BOOL	Status Check Start	xFBa1
BOOL	Feedback 1	xFBa2
BOOL	Feedback 2	Rel
BOOL	Release	Rel2

BOOL	Release 2	xAuto
BOOL	Extern Automatic	ACoHM
BOOL	Automatic Control Help Memory	ExCoHM
BOOL	Extern Control Help Memory	FBaOn
BOOL	feedback ON intern	FBaOff
BOOL	feedback OFF intern	FBaChange
BOOL	change extern feedback (0 FBa1=OFF FBa2=ON / 1 FBa1=ON FBa2=OFF)	FBa1Active

9.2.Free for Programmer-Specific Applications

BOOL	feedback 1 active	FBa2Active
BOOL	feedback 2 active	xAutoHM Si la entrada digital es 0 se activara la señal de alarma
BOOL	Extern Automatik Old	GAlQuitt Si la entrada digital es 1 se activa la señal de alarma
BOOL	General Alarm Quitt	Ign Si la entrada digital tiene estado 0, se comprueba el estado de la alarma
BOOL	ignore alarm	Sim Si la entrada digital tiene estado 1, se comprueba el estado de la alarma
BOOL	Simulation	Car Esta asignación esta incluida en el programa estandar
BOOL	automatic mode	MCo

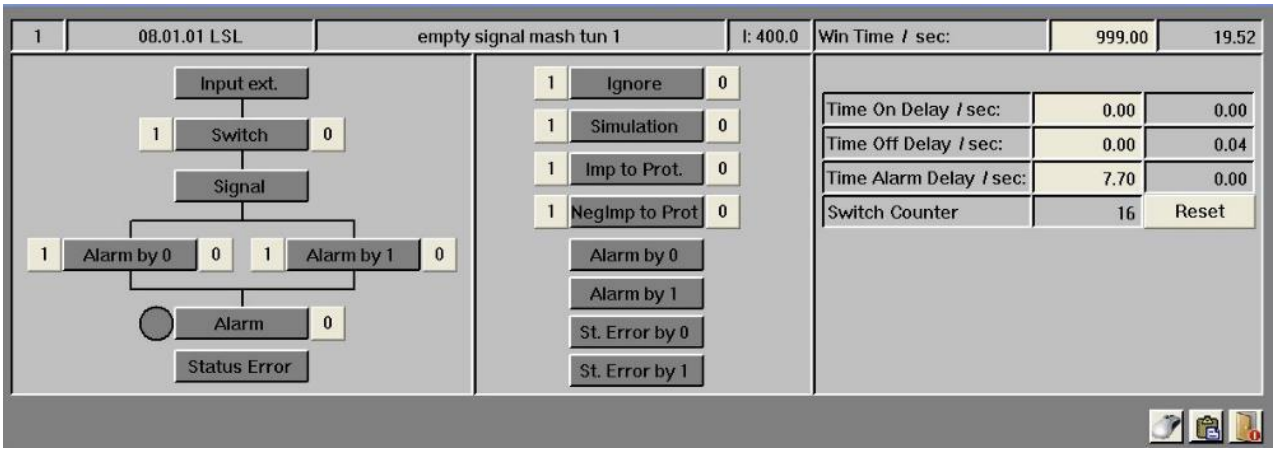
9.3.Digital Input Negative Pulse

BOOL	Manual Control
EmRel	BOOL El estado de la entrada digital es incorrecto
Emergency Release	InterlockGAl Alarma General fue o está activada Señal debe restablecerse por el operador (OK + Reset alarma)
BOOL	Interlock by alarm Estado incorrecto de la entrada digital para el próximo inicio del programa
Maint	BOOL Esta señal contiene los retardos programables, asi com la evaluación de un interruptor
Maintenence	Gal
BOOL	General Alarm

9.4.Alarm Delay

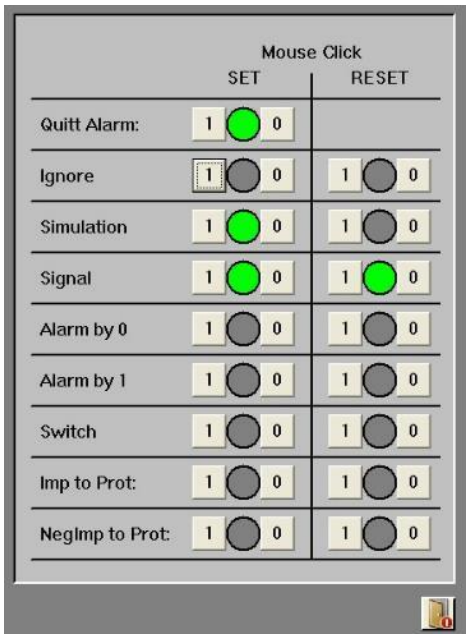
Gals	BOOL
General Alarm Save	SCE
BOOL	Status Check Error Si se envía un mensaje de error o no
Mov	BOOL El feedback se genera internamente, el Estado es siempre OK

actuator is moving for visu	On Si la señal de la entrada digital es 0 se activa una alarma
BOOL	actuator is ON Si la señal de la entrada digital es 1 se activa una alarma
Off	BOOL Con pulso positivo para registrar en el reporte de operaciones manuales
actuator is OFF	Out Con pulso negativo para registrar en el reporte de operaciones manuales
BOOL	Output Con cada flanco positivo se alterna el estado de la señal interna .Sig
User	BOOL Si la entrada tiene la señal física 1, se maniente señal interna por xxx segundos
Free for User Programm	TOnVal Si la entrada tiene la señal física 0, se maniente señal interna por xxx segundos
REAL	turn on delay value Si el estado de señal es incorrecto, se retrasa xxx segundos en disparar la alarma



9.5.Special Configurations

In addition to the system window for digital input for default parameter settings, there is a window for mouse settings. This determines what should happen when you click the mouse over the item.



- The digital signal of the sensor can be simulated in case of problems (non-process-critical signal, such as a transport sensor).
- Ignore alarms.

9.6.Programming Examples

Signal Query

```
U "Din.Din[19]. Sig"        Empty signal
S "PhaseEnd"                    Finish Step
```

Alarm Assessment

```
A "Step0"                   Stage is not 0
S "DIn". DIn[18]. EA0           Activate Alarm Signal 0

U "DIn". DIn[18]. GAlS"         Alarm
S "HoldReq"                 Maintains Unity
```

Status check

```
U "PA"                          Active Step
S "DIn". DIn[12]. SCS0          Status Error Triggered with Signal 0

A "DIn". DIn[12]. SCE"          No Error Status
S "PhaseEnd"                    Finish Step
```

DIn Assignment

```
U E 400.0                   Physical Entry Address
= "DIn". DIn[1].xSig   Active Program DiN 1 Signal
U E 599.7                   Physical Entry Address
= "DIn". DIn[1600].xSig     Active Program Signal DIn 1600
```

10. Analog Input (AIn)

Also when analog input is any analog input signal with a measurement result, but calculating values or values entered manually.
A calculated value, for example, is a capacity of a tank. The pressure difference is first calculated on the basis of this differential pressure of the tank volume on a base hl. It is not possible to display this content currently in the image of an analog input, it is only displayed in final value.
The Analog input associated with the physical input is done on the FC 503 "TransAIn". (Example in "AIn allocation").
For calculated values or analog input numbers, they should be used for a map with PEW for existing reservations in the hardware block.

10.1. Process Value Intern (without Polygon)

TOnSp	REAL	turn on delay setpoint
TOfVal	REAL	Turn Off Delay Value
TOfSp	REAL	Turn Off Delay Setpoint
ADVal	REAL	alarm delay value
Adsp	REAL	alarm delay setpoint
TInterlock	REAL	time interlock before restart
SwCntVal	DINT	Switch Counter Value
RunTimeVal	DINT	duty timer value (seconds)
B31	Allocation	Default
Comment	Aco	0
Aimed at actuators	ExCo	0
External control, e.g. by a switch (digital input)	SCS	0
Start the Health Check Query	xFBa1	x
FeedBack 1	xFBa2	x
FeedBack 2	Rel	1
Safety Locks	Rel2	1
Production	xAuto	0
Automatic signal (usually "RUN" accompanying the Unit)	User	x
Free for Programmer-Specific Applications	BOOL	Allocation
Comment	FBaOn	Actuator On Feedback (Energized)
FBaOff	Actuator Feedback Off (Non-Energized)	Gal
Alarm General	Gals	Stores General Alarm
SCE	Indicates an error condition	Mov
The actuator is about to move	On	Actuator On (Output & Feedback On & No Feedback Off)
Off	Actuator off (no output and feedback off)	Out
Exit	BOOL	general alarm
Allocation	Comment	FBaChange
Feedback	FBa1Active	Feedback 1 is present

FBa2Active	Feedback 2 is present	GAlQuitt
Recognizing an Alarm (Gals Reset)	Ign	Ignore Alarm
Sim	Simulation Mode	Car
Auto Mode	MCo	Manual control
EmRel	Emergency Unlock	InterlockGAl
The	Maint	Actuator under maintenance
TOnSp	Delay Switching	TOfSp
Delay to disconnection	Adsp	Fault Monitoring Delay
TInterlock	Timeout for reconnection	high scaling
LLAVal	REAL	low low alarm value
LLVal	REAL	low low value (warning limite)
LVal	REAL	low value
HVal	REAL	high value
HHVal	REAL	high high value (warning limite)
HHAVal	REAL	high high alarm value
LLAHys	REAL	low low alarm hysteresis
LLHys	REAL	low low hysteresis
LHys	REAL	low hysteresis
SpHys	REAL	setpoint hysteresis
HHys	REAL	high hysteresis
HHHys	REAL	high high hysteresis
HHAHys	REAL	high hgih alarm hysteresis
ADVal	REAL	alarm delay value
ADSp	REAL	alarm delay setpoint
PoTNo	REAL	positive = polygon table number / negative = offset
xPVal	REAL	raw value from extern
iPVal	REAL	process value intern (without polygon)

10.2. Process Value

Allocation	Guy	Comment
EA0	BOOL	enable alarm by 0-signal Al caer por debajo de los límites de alarma Bajo Bajo, se activa una alarma
EA1	BOOL	enable alarm by 1-signal Al sobrepasar el limite de alarma alta alta se activa una alarma
SCS0	BOOL	status check alarm by 0-signal Si esta señal 1 se dispara como una alarma
SCS1	BOOL	status check alarm by 1-signal En este caso, el valor es llevado a xPVal No hay conversión de escala baja o escala Alta
xSig	BOOL	signal extern
B29	BOOL	spare Normalmente, esta variable se suministra desde el programa estándar Si no hay periferia, por lo que este valor debe ser suministrado por el programador

10.3. Process Value

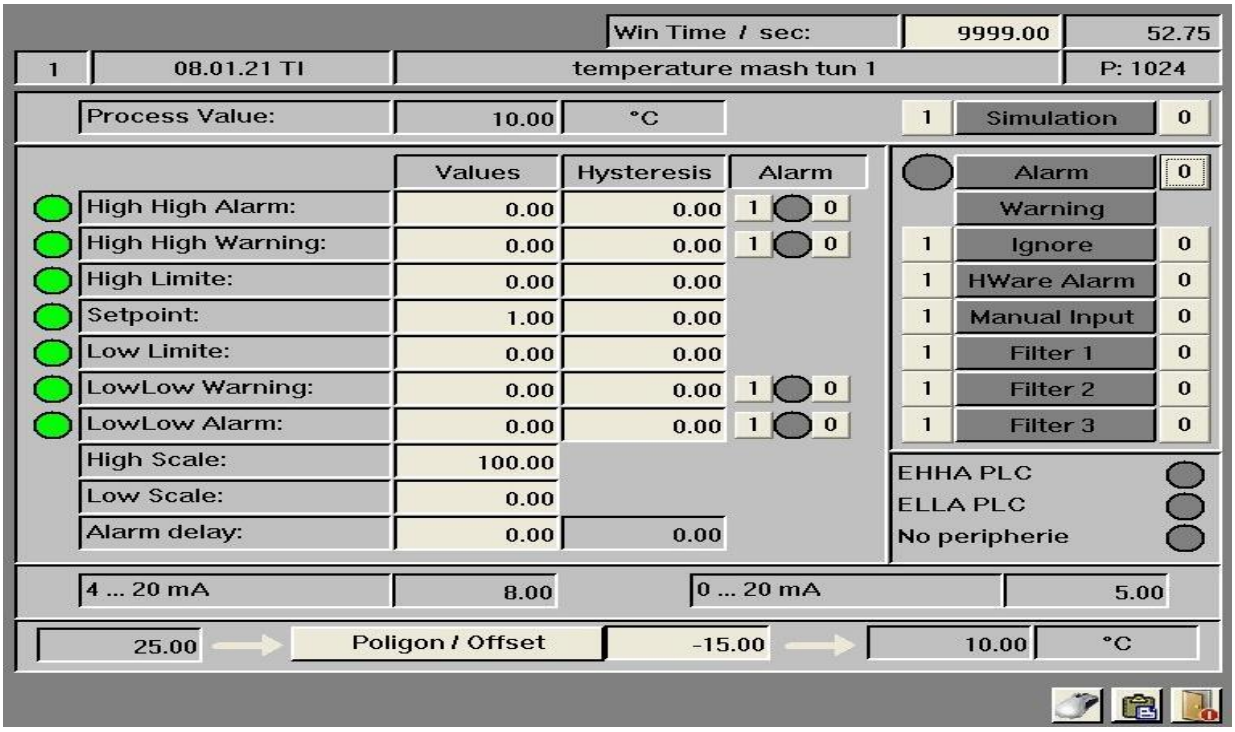
B30	BOOL

spare	B31 PVal > LLAVal => 1-Señal A este valor se asigna una histéresis
BOOL	spare PVal > LLVal => 1-Señal A este valor de asigna una histéresis
AlHM	BOOL PVal > LVal => 1-Señal A este valor se asigna una hitéresis
help memory for alarm	ImpHM PVal > SP => 1-señal A este valor se asigna una hitéresis
BOOL	help memory for impulse PVal > HVal => 1-Signal A este valor se asigna una hitéresis
xSigHM	BOOL PVal > HHVal => 1-Signal A este valor se asigna una hitéresis
Signal Extern Help Memory	B19 PVal > HHAVal => 1-Signal A este valor se asigna una hitéresis
BOOL	spare Cae por debajo del límite mínimo (generalmente 2 mA) o superan el límite máximo (normalmente 22 mA)
B20	BOOL Estado de la entrada analógica no está bien
spare	B21 Alarma General fue o está activada Señal debe restablecerse por el operador (OK + Reset alarma)
BOOL	spare Si sólo con fines de visualización o para información de la entrada en el sistema.
B22	BOOL El valor convertido en la unidad física

10.4. Positive Value = Conversion Value Polygon Table Number

spare	B23
BOOL	spare
GAlQuitt	BOOL Si se envía un mensaje de error o no
General Alarm Quitt	Ign El valor “PVal” es manipulado directamente en la visualizacion
BOOL	ignore alarm Cuando los valores de entrada fisica está por debajo o sebre el límite
Sim	BOOL se genera una alarma si "PVal" es menor que "LLAVal"
Simulation	iEA0 se genera una alarma si "PVal" es mayor que "HHAVal"
BOOL	intern alarm by 0 se genera una alarma si "PVal" es menor que "LLAVal"
iEA1	BOOL se genera una advertencia si "PVal" es mayor que "HHAVal"

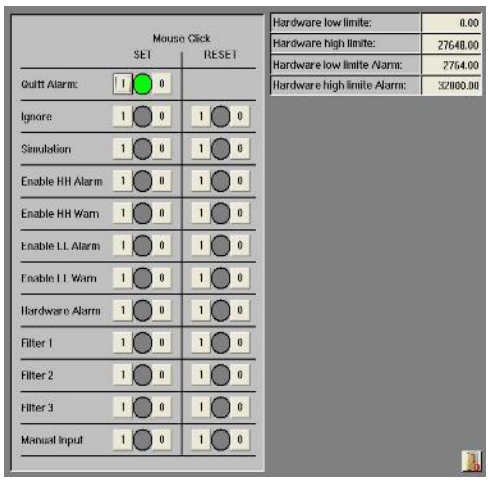
intern alarm by 1	ImpProt El valor de proceso se filtra. Tiene sentido en el caso de entradas oscilante
BOOL	write impule flank to protocol El valor de proceso se filtra. Tiene sentido en el caso de entradas oscilante
ImpNegProt	BOOL El valor de proceso se filtra. Tiene sentido en el caso de entradas oscilante
write negative impule flank to protocol	Switch La entrada analógica esta presente virtualmente. El valor de proceso lo ingresa el operador
BOOL	convert as switch output
Gal	BOOL Valor físico mínimo de la entrada analógica Se utiliza para convertir de unidades de escala (mA) en la unidad física
General Alarm	Gals Valor físico máximo de la entrada analógica Se utiliza para convertir de unidades de escala (mA) en la unidad física
BOOL	General Alarm Save Se genera una alarma si cae del limite bajo bajo
SCE	BOOL Se genera una adevertenca si cae del limite bajo bajo
Status Check Error	Gis
BOOL	signal state
Imp	BOOL Se genera una adevertenca si sobrepasa el limite Alto Alto
Impulse Flank	ImpNeg Se genera una alarma si sobrepasa el limite Alto Alto
BOOL	Negative Impulse Flank
B06	BOOL
spare	User
BOOL	free for user
TOnDVal	REAL
turn on delay value	TOnDSp
REAL	turn on delay setpoint
TOfDVal	REAL Si el estado es incorrecto se genera una alarma después de xxx segundos
Turn Off Delay Value	TOfDSp Valor negativo= desplazmiento del valor de proceso



10.5. Special Configurations

In addition to the system window for analog input for default parameter settings, there is a window for mouse settings. This determines what should happen when you click the mouse over the item.

In addition to the mouse parameterization, you can determine the overall scale of the inputs:



- Low Limit Hardware – Scale Division at 4 mA (0 mA)
- High Limit Hardware – Scale Division at 20 mA
- Hardware Alarm Low Limit – If the analog input falls below this value, the wire break alarm is triggered.
- Hardware Limit Alarm High – If the analog input exceeds this value, the overflow alarm is triggered.

10.6. Programming Examples

Process Value Transfer

```
L "AIn". AIn[4]. PVal        Temperature Measurement
T "U002". For[12]. Val            Unit Parameter 12
```

Alarm Assessment

```
A "Step0"                    Stage is not 0
```

```
S "AIn". AIn[18]. SHE          Activate the low low limit alarm

U "AIn". AIn[18]. GAlS"        Alarm
S "HoldReq"                    Maintains Unity
```

AIn Assignment

```
L PEW 1024                     We charge the input process value
ITD                            Convert to decimal
DTR                            We convert to real
T "Bx AIn D". AIn[1].xPVal     we transfer to the process value of the Ain
L PEW  1982                    We charge the value of the process
ITD                            Convert to decimal
DTR                            We convert to real
T "Bx AIn D". AIn[480].xPVal   we transfer to the process value of the Ain
```

11. PID Regulator (PID)

Each analog output is supplied by a PID controller, but you don't always have a PID for analog outputs. An example of a PID without analog output would be a cascade control or a pulse output control.
The PID associated with the physical output is performed on the FC 504 "TransPID". (Example in "PID assignment").
For cascade control outputs or pulse PID numbers, you must use the one that exists on the corresponding map and lock the existing reservations in the hardware.

11.1. derivative part

REAL	Turn Off Delay Setpoint	ADVal
REAL	alarm delay current value	Adsp
REAL	alarm delay setpoint	SwCntVal
DINT	Switch Counter Value	static output value
MStrt	Allocation	Default
Comment	EA0	0
Alarm at 0 – Signal	EA1	0
Alarm at 1 – Signal	SCS0	0
Check the alarm status at 0 – Signal	SCS1	0
Check the alarm status at 1 – Signal	xSig	x
Physical Input Signal	User	x
Free for Programmer-Specific Applications	BOOL	spare
Allocation	Comment	Gal
General Alarm	Gals	Stores General Alarm
SCE	Points out an error condition Incorrect	Gis
Signal Status	Imp	Digital Input Positive Pulse
ImpNeg	Digital Input Negative Pulse	warning
GAlQuitt	Bezeichnung	Kommentar
GAlQuitt	Recognizing an Alarm (Gals Reset)	Ign
Ignore Alarm	Sim	Simulation Mode
iEA0	Alarm on signal 0	iEA1
Alarm on signal 1	ImpProt	Enter Positive Edge for Manual Operations
ImpNegProt	Enter Negative Flank for Manual Operations	Switch
Evaluation of the switch as a digital input	TOnDSp	Delay signal change from 0 to 1
TOfDSp	Delay signal change from 1 to 0	Adsp
Alarm Delay	BOOL	general alarm
GAlS	BOOL	general alarm save
SCE	BOOL	status check error
Filter1	BOOL	filter 1 on (75%)
Filter2	BOOL	filter 2 on (88%)
Filter3	BOOL	filter 3 on (94%)

CA	BOOL	control acting (1 = inverse)
User	BOOL	memory free for user
OVal	REAL	output value
Sp	REAL	setpoint
PVal	REAL	process value
xSp	Assignment	Guy
Comment	SHE	BOOL
enable low low alarm	EHHA	BOOL
enable high high alarm	xAl	BOOL
alarm from extern	NPA	BOOL
No peripherie adaption	B28	BOOL
spare	B29	BOOL
spare	B30	BOOL
spare	B31	BOOL
spare	MLLA	BOOL
Low Low Alarm - Alarm if enabled	MLL	BOOL
Low Low Limit - warning if enabled	ML	BOOL
low limit	Msp	BOOL
setpoint	MH	BOOL
High Limit	MHH	BOOL
High High Limite - warning if enabled	MHHA	BOOL
High High Alarm - Alarm if enabled	MHWA	BOOL
alarm from hardware	GAlQuitt	BOOL
General Alarm Quitt	Ign	BOOL
ignore alarm	Sim	BOOL
Simulation	iEHWA	BOOL
Enable Hardeware Alarm	iELLA	BOOL
enable LL alarm	iEHHA	BOOL
enable HH alarm	iELLW	BOOL
enable LL warning	iEHHW	BOOL
enable HH warning	Gal	BOOL
General Alarm	Gals	BOOL
General Alarm Save	Warn	BOOL
General warning	Filter1	BOOL
filter 1 on (75%)	Filter2	BOOL
filter 2 on (88%)	Filter3	BOOL
filter 3 on (94%)	ManuInp	BOOL
Manual Input (no peripherie)	User	BOOL
Memory Free for User	PVal	REAL
Process Value	Sp	REAL
setpoint	LScal	REAL

11.2. Fuzzy Variable 2

Low scaling	HScal	REAL
High Scaling	LLAVal	REAL
Low Low Alarm Value	LLVal	REAL
Low Low Value (Warning Limite)	LVal	REAL
Low Value	HVal	REAL Hay dos maneras de comenzar el regulador 1. Consigna inicio controlador de tiempo = 0 el valor de salida inicial se calcula una vez (Error * KP + inicia la producción de valor) 2. Controlador de consigna Inicio> 0 se escribe en la salida del PID, siempre y cuando haya transcurrido el tiempo de inicio del valor inicial parametrizado.
High Value	HHVal	REAL
High High Value (Warning Limite)	HHAVal	REAL
High High Alarm Value	LLAHys	REAL
Low Low Alarm Hysteresis	LLHys	REAL
Low Low Hysteresis	LHys	REAL Es el punto de consigna usado por el programa, es ajustado por este valor
Low Hysteresis	SpHys	REAL Se utiliza en valor de proceso
Setpoint Hysteresis	HHys	REAL En ciertos casos, este valor tiene que ser proporcionada en el programa Normalmente, la posibilidad de parametrización a través de la Visu es suficiente.
High Hysteresis	HHHys	REAL Integra una logica difusa simple, la variable se adecua influyendo en las maniobras de control.
High High Hysteresis	HHAHys	REAL Integra una logica difusa simple, la variable se adecua influyendo en las maniobras de control.

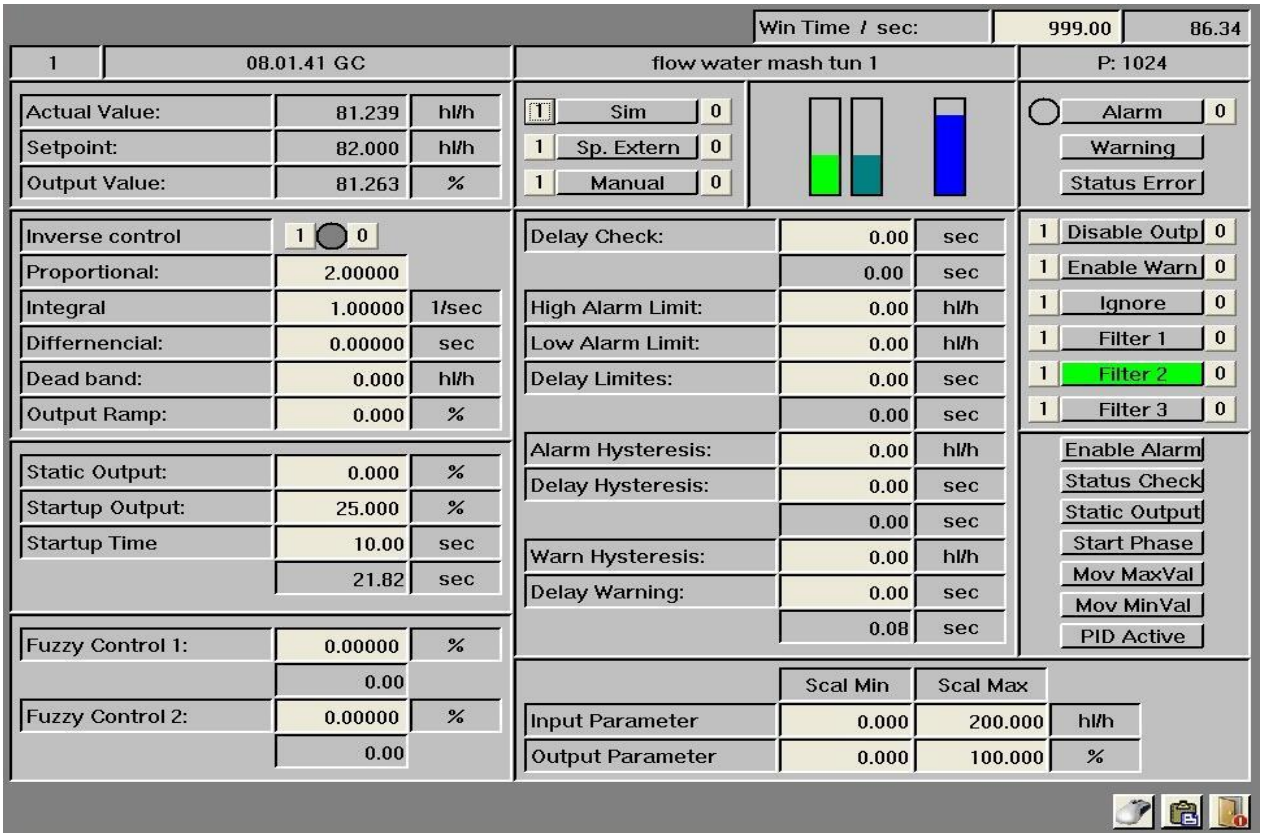
11.3. Indicates an error condition of

High HGIH Alarm Hysteresis	ADVal
REAL	alarm delay value El estado del PID no es correcto
Adsp	REAL Alarma General fue o está activada Señal debe restablecerse por el operador (OK + Reset alarma)
alarm delay setpoint	PoTNo Estado incorrecto del PID para el próximo inicio del programa

11.4. Fuzzy Gain 2

REAL	positive = polygon table number / negative = offset
xPVal	REAL
Raw value from extern	iPVal Si se envía un mensaje de error o no
REAL	Process Value Intern (without Polygon) El valor “PVal” es manipulado directamente en la visualizacion
MCOn	Allocation Si la función de control se desactiva, el valor de salida se puede escribir directamente
Default	Comment El valor de entrada de consigna se uede ingresar en la visualización
SHE	0 Cuando la salida está desactivada, el valor de salida PID no está transmitiendo a la periferia
Low Low Alarm Activation	EHHA Si el valor de proceso está fuera de los límites se genera con la advertencia
0	High High Alarm Activation When the high high alarm limit is exceeded, an alarm is triggered
xAl	0 El valor de proceso se filtra. Tiene sentido en el caso de entradas oscilante
External Alarm	NPA El valor de proceso se filtra. Tiene sentido en el caso de entradas oscilante
0	No Peripherals In this case, the value is brought to xPVal
User	x Si activa la función de mando, el valor de salida se puede introducir directamente.
Free for Programmer-Specific Applications	xPVal Se puede controlar externamente la consigna, la consigna se puede introducir directamente
x	Process Value Normally, this variable is supplied from the standard program If there is no periphery, so this value must be supplied by the programmer
LScal	Escala Baja Valor mínimo física de la entrada analógica Sólo para la visualización de tendencia, no tiene ningún efecto sobre la función del controlador
Allocation	Comment Valor máximo física de la entrada analógica Sólo para la visualización de tendencia, no tiene ningún efecto sobre la función del controlador
MLLA	Alarm Limits Under PVal > LLAVal => 1-Signal
MLL	Low Limit Low PVal > LLVal => 1-Signal
ML	Low Limit PVal > LVal => 1-Signal A hysteresis is assigned to this value
Msp	Desired value OK

	PVal > SP => 1-signal This value is assigned a hitheresis En una referencia de tiempo de 0 segundos para la fase de puesta en marcha, este valor se utiliza como una compensación al cálculo inicial de la salida del controlador.
MH	High Limit PVal > HVal => 1-Signal This value is assigned a hitheresis
MHH	High Limit High PVal > HHVal => 1-Signal This value is assigned a hitheresis
MHHA	High Limit Alarm High PVal > HHAVal => 1-Signal This value is assigned a hitheresis
MHWA	Hardware alarm (4-20 mA) Falls below the minimum limit (usually 2 mA) or exceeds the maximum limit (typically 22 mA)
Gal	Alarm General Analog input status is not correct
Gals	Stores General Alarm General Alarm was or is activated
Warn	Operator Warning : If only for display purposes or for system entry information.
PVal	Process Value The value converted to the physical unit
WDSp	Allocation En ese momento, una advertencia puede ser retrasado debido a superar la histéresis.
Comment	GAlQuitt
Recognizing an Alarm (Gals Reset)	Ign
Ignore Alarm	Sim
Simulation Mode	iEHWA Cambio máximo en la salida por segundo a punto de ajuste = 0, este valor está inactivo
Triggers Hardware Monitoring Alarm	iELLA Siempre y cuando el valor de proceso está dentro de la banda muerta, la salida del regulador se congela
Triggers Low Limit Alarm	iEHHA Ponderación la primera lógica difusa
High High Limit Alarm Trigger	iELLW Ponderación la primera lógica difusa



11.5. Special Configurations

In addition to the system window for the PID for default parameter settings, there is the window for mouse settings. This determines what should happen when you click the mouse over the item.

In addition to the mouse parameterization, you can determine the overall scale of the inputs:

- Output value 0% – dissipation on output card, 0% PID output
- Output value 100% – dissipation on output card, 100% PID output

11.6. Programming Examples

Transfer of Securities

L "Uxx". For[12]. Val	Parameter Unit 12 – Temperature Measurement
L "PID". PID[4].xPVal	Process Value for PID
L "Uxx". For[12]. Sp	Parameter Unit 12 – Temperature Measurement
L "PID". PID[4].xSp	Nominal value of the PID

Alarm Assessment

U "Act.Act[45]. Out"	Actuator Output
A "CIP"	CIP
S "PID". PID[4]. Eal	Enable Low Limit Alarm
U "PID". PID[4]. GAlS"	Alarm
S "HoldReq"	Maintains Unity

Home / Static Output of the PID

```
U "Act.Act[45]. Out"        Actuator Output
U "CIP"                         CIP
S "PID". PID[4]. MStC           Enable Static Output

U "Act.Act[45]. Out"        Actuator Output
S "PID". PID[4]. MStrt          Start the PID
```

PID Assignment

```
L "PID". PID[1]. Oval       PID output value 1
L #OutFactor                    are 327.67
*R                              multiplier
RND                             Rounds
T PAW 1024                  Transfer the value to the physical output
L "PID". PID[480]. Oval         PID 480 output value
L #OutFactor                    are 327.67
*R                              multiplier
RND                             Rounds
T PAW 1982                  Transfer the value to the physical output
```

12. Counter Module (Cnt)

The Counter (Count) Module is used to add certain values of pulses in the cyclic sequence. Up to 320 meters are available for the PLC, which is more than enough. A fixed assignment of peripherals and counter module number does not exist here.
It is often useful to split a flow meter in the hardware into several counter modules in the software. An example would be a water flowmeter. In order to calculate the amount of water that was pumped into vessel 1, in addition to integrating an online query, the cleanest solution is to co-figure a meter for each line.
The value that each pulse delivered to the counter signifies must also be adjusted.

12.1. counter value

Triggers Low Limit Warning An	iEHHW	High High Limit Warning Active
Filter1	Filter 1 in (75%)	Filter2
Filter 2 in (88%)	Filter3	Filter 3 in (94%)
ManuInp	Manual (non-peripheral) inputThe	Sp
Setpoint value	LScal	Low Scaling
HScal	Scaling High	LLAVal
Low Low Limit Alarm	LLVal	Low Low Limit Warning
LVal	Low Limit	HVal
High Limit	HHVal	Warning High High
HHAVal	Alarm High High	LLAHys
Hysteresis for Low Limit Alarm	LLHys	Hysteresis for Low Limit Warning
LHys	Low Limit Hysteresis	SpHys
Setpoint hysteresis	HHys	High limit hysteresis
HHHys	Hysteresis for High Limit Adevity	HHAHys
Hysteresis for High High Limit Alarm	Adsp	Alarm Delay:
PoTNo	Positive Value = Conversion Value Polygon Table Number	high high alarm - alarm if enabled
ImpHM	BOOL	impulse help memory
GAlQuitt	BOOL	general alarm quitt
Ign	BOOL	ignore alarm
Sim	BOOL	simulation
Reset	BOOL	reset counter
iELLA	BOOL	counting reserve
iEHHA	BOOL	enable HH alarm
iELLW	BOOL	enable LL warning
iEHHW	BOOL	enable HH warning
Gal	BOOL	general alarm
GAlS	BOOL	general alarm save
Warn	BOOL	general warning
Imp	BOOL	impulse flank
B04	Allocation	Guy
Comment	Eal	BOOL
enable alarm	SCS	BOOL
Status Check Start	MStC	BOOL

static output value	MStrt	BOOL
starting value	MOVMin	BOOL
output value min.	MOVMax	BOOL
output value max.	OVOn	BOOL
output value on	B31	BOOL
spare	B16	BOOL
spare	B17	BOOL
spare	B18	BOOL
spare	B19	BOOL
spare	AlHM	BOOL
help memory for alarm	AHystHM	BOOL
Help Memory Outside Hysterese	StrtHM	BOOL
Help Memory Starting Value Active	Warn	BOOL
warning	GAlQuitt	BOOL

12.2. Free for Programmer-Specific Applications

General Alarm Quitt	Ign	BOOL
ignore alarm	Sim	BOOL Se monitorea si en algún momento se ha registrado un pulso Si no, se activa una alarma
Simulation	MCOn	BOOL Al caer por debajo de los límites de alarma Bajo Bajo, se activa una alarma
mode controller on (0=off)	MSpExt	BOOL Al sobrepasar el limite de alarma alta alta se activa una alarma
mode setpoint extern (0=intern)	DisOut	BOOL Si esta señal 1 se dispara como una alarma
disable output peripherie (0=enable)	EW	BOOL Si esta señal esta en 1, el contador no se puede resetear
enable warning	B15	BOOL A esta señal de entrada digital del hardware se le asiga a un contador
spare	Gal	BOOL

12.3. Process Value

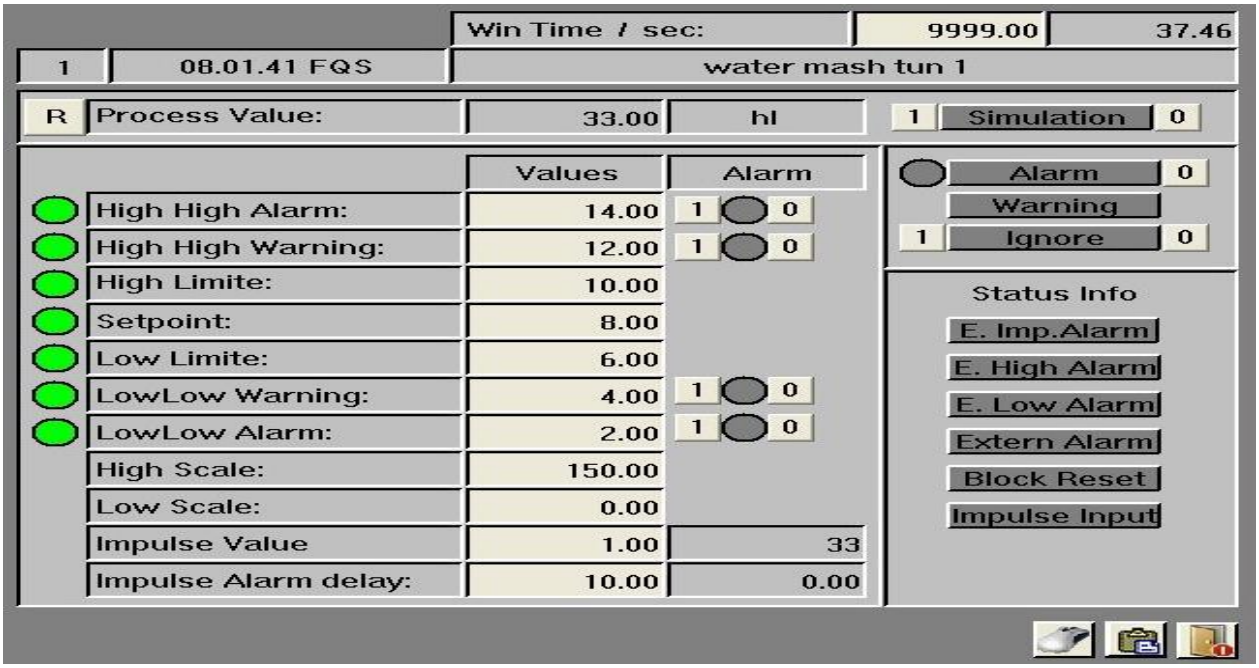
General Alarm	Gals
BOOL	General Alarm Save PVal > LLAVal => 1-Señal A este valor se asigna una histéresis
SCE	BOOL PVal > LLVal => 1-Señal A este valor de asigna una histéresis
Status Check Error	Filter1 PVal > LVal => 1-Señal A este valor se asigna una hitéresis
BOOL	filter 1 on (75%) PVal > SP => 1-señal A este valor se asigna una hitéresis
Filter2	BOOL

	PVal > HVal => 1-Signal A este valor se asigna una hitéresis
filter 2 on (88%)	Filter3 PVal > HHVal => 1-Signal A este valor se asigna una hitéresis
BOOL	filter 3 on (94%) PVal > HHAVal => 1-Signal A este valor se asigna una hitéresis
AC	BOOL Estado de la entrada analógica no está bien
control acting (1 = inverse)	User Alarma General fue o está activada Señal debe restablecerse por el operador (OK + Reset alarma)
BOOL	Memory Free for User Si sólo con fines de visualización o para información de la entrada en el sistema.
Oval	REAL Un flanco positivo se forma a partir de la señal xSig.
Output Value	Sp

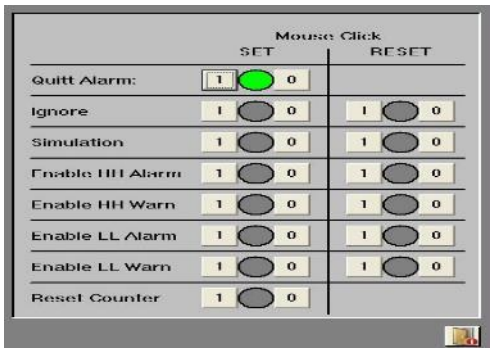
12.4. Value of a pulse

REAL	setpoint
PVal	REAL
Process Value	xSp Si se envía un mensaje de error o no
REAL	Setpoint Extern El valor “PVal” es manipulado directamente en la visualizacion
xPVal	REAL El contador se pone a cero
Process Value from User Programm	LScal se genera una alarma si "PVal" es menor que "LLAVal"
REAL	Low scaling se genera una alarma si "PVal" es mayor que "HHAVal"
HScal	REAL se genera una alarma si "PVal" es menor que "LLAVal"
High Scaling	OVMin se genera una advertencia si "PVal" es mayor que "HHAVal"
REAL	output value min.
OVMax	REAL Valor físico mínimo de la entrada analógica Se utiliza para convertir de unidades de escala (mA) en la unidad física
output value max.	Stc Valor físico máximo de la entrada analógica Se utiliza para convertir de unidades de escala (mA) en la unidad física
REAL	static output value % Se genera una alarma si cae del limite bajo bajo
Strt	REAL Se genera una adevertenca si cae del limite bajo bajo
starting value %	StrTVal
REAL	starting time value
StrTSp	REAL Se genera una adevertenca si sobrepasa el limite Alto Alto

starting time setpoint	LLAVal Se genera una alarma si sobrepasa el limite Alto Alto
REAL	low value for alarm Si el estado es incorrecto se genera una alarma después de xxx segundos.
HHAVal	REAL No se introduce el valor de un pulso de la hardware Ejemplo, peso de malta: 1 impulso = 50 kg; Así ImpVal = 50



12.5. Special Configurations



In addition to the system window for the counter for default parameter settings, there is also a window for mouse settings. This determines what should happen when you click the mouse over the item.

12.6. Programming Examples

Signal Transfer

U "Act". Act[18]. Out — Water Valve
U "DIn". Din[233]. Imp — Positive Ditital Inlet Flank
= "Cnt". Cnt[4].xSig — External signal for the counter

Transfer of Securities

L "Cnt". Cnt[4]. PVal — Water Meter
T "Uxx". Para[17]. Val — Unit Parameter 17

Alarm Assessment

U "Act". Act[18]. Out Water Valve
S "Cnt". Cnt[4]. EAImp Enable Pulse Alarm

U "Cnt". Cnt[4]. GAIS" Alarm
S "HoldReq" Maintains Unity

13. Message Module (Msg)

Since the other modules such as actuators, digital inputs, etc. already integrate messages, this message module is very rarely used. It is only used to generate alarms or instructions for the operator that are not directly related to the periphery.

13.1. alarm delay setpoint

High value for alarm	AHys	REAL
Hysteresis Band for Alarm	CheckDVal	REAL
Check delay value	CheckDSp	REAL
Check Delay Setpoint	ADHLVal	REAL
alarm delay high low limte value	ADHLSp	REAL
alarm delay high low limit setpoint	ADVal	REAL
Alarm Delay Hysteresis Value	Adsp	REAL
Alarm Delay Hysteresis Setpoint	WHys	REAL
Hysteresis Band for Warning	WDVal	REAL
warning delay hysteresis value	WDSp	REAL
warning delay hysteresis setpoint	KP	REAL
propotional gain (unitless)	KI	REAL
Integral Gain (1/sec)	KD	REAL
Derivative Gain (SEC)	RampV	REAL
ramp value for OVAL (per second)	DeadB	REAL
Dead Band For Error	Fuzzy1G	REAL
fuzzy 1 gain	Fuzzy1V	REAL
Fuzzy 1 Variable	Fuzzy1VOld	REAL

Fuzzy 1 Variable Old	Fuzzy2G	REAL
fuzzy 1 gain	Fuzzy2V	REAL
Fuzzy 1 Variable	Fuzzy2VOld	REAL
Fuzzy 1 Variable Old	iOVal	REAL
output value intern	And	REAL
Control Error	DPart	REAL
derivative part	BOOL	spare
GAl	Attachment	Default
Comment	Eal	0
Alarm Activation Evaluation	SCS	0
Turn on Health Check	MStC	0
Slider Output Setting to Parameterize Static Output Value (Read-Only)	MStrt	0
There	MOVMin	0
Adjustment of the Output of the slider to parameterize the minimum base	MOVMax	0
Maximum Output Value	OVOn	0
Slider Output Setting to Parameterize a Maximum Output Value	User	x
Free for Programmer-Specific Applications	xSp	x

13.2. Free for Programmer-Specific Applications

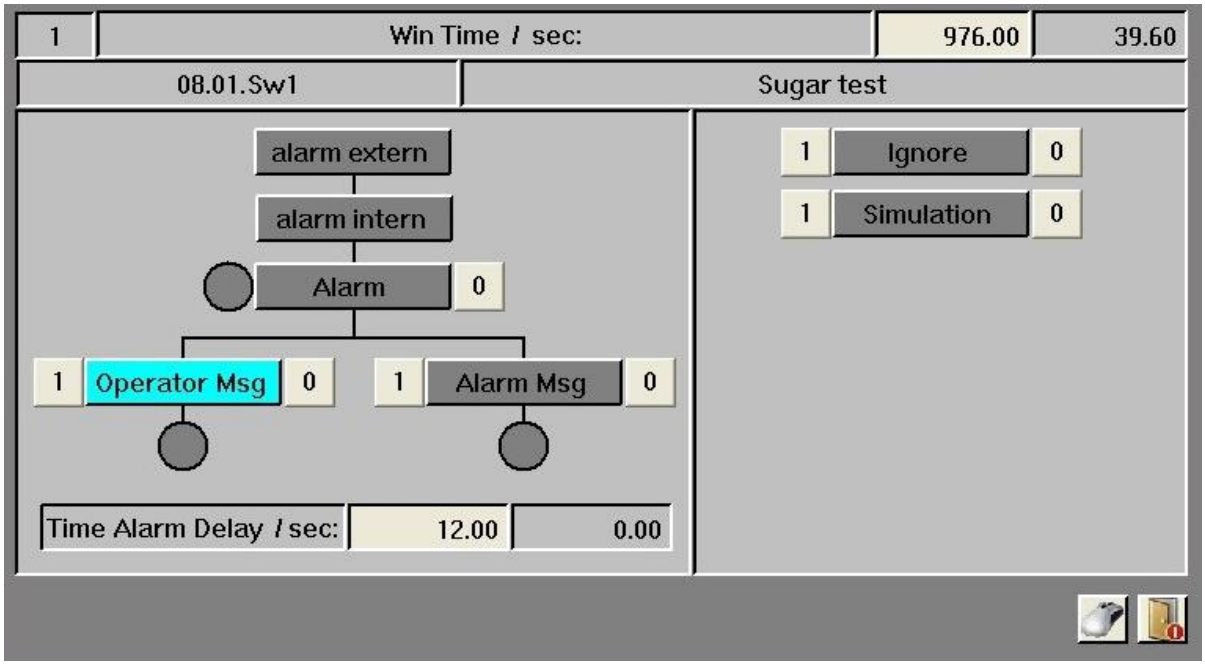
External Set Point	xPVal	x
External Process Value	Stc	x Si esta señal es 1,se genera un mensaje (o alarma) de operador
In	Fuzzy1V	x

13.3. Alarm message active

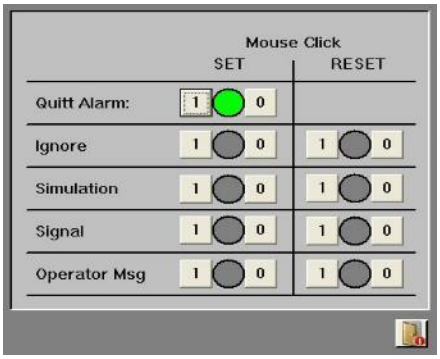
Fuzzy Variable 1	Fuzzy2V
x	Fuzzy Variable 2 Integrates a simple fuzzy logic, the variable adapts influencing the control maneuvers.
GAlS	Assignment Alarma General fue o está activada Señal debe restablecerse por el operador (OK + Reset alarma)
Comment	Gal La nota operador está activo.
General Alarm PID	Gals La alarma está activa.

13.4. Alarm Delay (message)

Stores General Alarm	SCE
Indicates an error condition of	Reconocimiento de una alarma (Restablecer Gals)
Allocation	Comment Si se envía un mensaje de error o no
GAlQuitt	Recognizing an Alarm (Gals Reset) El valor “PVal” es manipulado directamente en la visualizacion
Ign	Ignore Alarm Whether an error message is sent or not En OPMsg = 0, se genera un mensaje de alarma cuando la señal es xAlarm activo
Sim	Simulation Mode The value "PVal" is manipulated directly in the display



13.5. Special Configurations



In addition to the system window for the counter for default parameter settings, there is also a window for mouse settings. This determines what should happen when you click the mouse over the item.

13.6. Programming Examples

Generate Message

U "Malzlaster Wartet"	Malta truck is waiting
= "Msg". Msg[12].xAlarm	External signal for the message

Alarm Assessment

U "Msg". Msg[12]. Gals	Active Message
S "SignalLamp"	Visual cue for the operator

14. Software Switch

In order to achieve simple and uniform operation, the Switch module (software switch) is integrated for the operator to generate a signal to the PLC (independent of the other modules), the default switch most commonly used would be the button to "confirm alarm" or to confirm a manual operation.

14.1. free for user

MCOn	Shutdown Control Function (1 = Shutdown)	MSpExt
External conigna point (1 = external)	DisOut	Disable Output (1 = Disabled)
EW	Trigger Warning Alarm	Filter1
Filter 1 in (75%)	Filter2	Filter 2 in (88%)
Filter3	Filter 3 in (94%)	AC
Control Direction (1 = Reverse)	Oval	PID Output Value
Sp	PID setpoint	PVal
Process Value	LScal	Low Scale
HScal	High Scale	OVMin
Minimum PID Output Value	OVMax	Maximum PID Output Value
Stc	Static Output Value (Read-Only)	Strt
Initial Value for Control	StrTSp	Controller Setup Time at Commissioning
LLAVal	Minimum Limit Alarm	HHAVal
Maximum Limit Alarm	AHys	Alarm hysteresis
CheckDSp	Alarm Activation Delay	ADHLSp
Alarm Delay When Limits Are Exceeded	Adsp	Alarm hysteresis delay is overcome
WHys	Hysteresis Warning	WDSp
At	KP	Control provides
KI	Comprehensive control	KD
Differential Control	RampV	On-ramp
DeadB	As	Fuzzy1G
Fuzzy Gain 1	Fuzzy2G	Fuzzy Gain 2
B13	BOOL	spare
B14	BOOL	spare
B15	BOOL	spare
B00	BOOL	spare
B01	BOOL	spare
B02	BOOL	spare
Sig	BOOL	spare
B04	BOOL	spare
B05	BOOL	spare
B06	BOOL	spare
User	BOOL	free for user

14.2. Free for Programmer-Specific Applications

Asignación	Default	Comentario
Set	0	Sirve como un interruptor en el programa de secuencia Mientras esta señal es 1, también se ofrece en la visualización, que el interruptors se puede operar.
Reset	0	Sirve como un interruptor para retroceder en el programa de secuencia Mientras esta señal es 1, también se ofrece en la visualización, que el interruptor no se puede operar.
User	x	Libre para aplicaciones específicas del programador

14.3. Condition Software Switch Status

Allocation	Guy
Comment	EAImp Estado del interruptor de software

14.4. Programming Examples

Switch reboot

```
A "RUN"
S "Switch". Switch[3]. Reset        Reset the switch to block operations
```

Checking the status of the Switch

```
U "PH"
U "Switch". Switch[3]. Gis          Confirmation of the operator "Manual Sugar Emptying"
S "Act". Act[42]. Aco               Mixer
```

15. Unidad (unit)

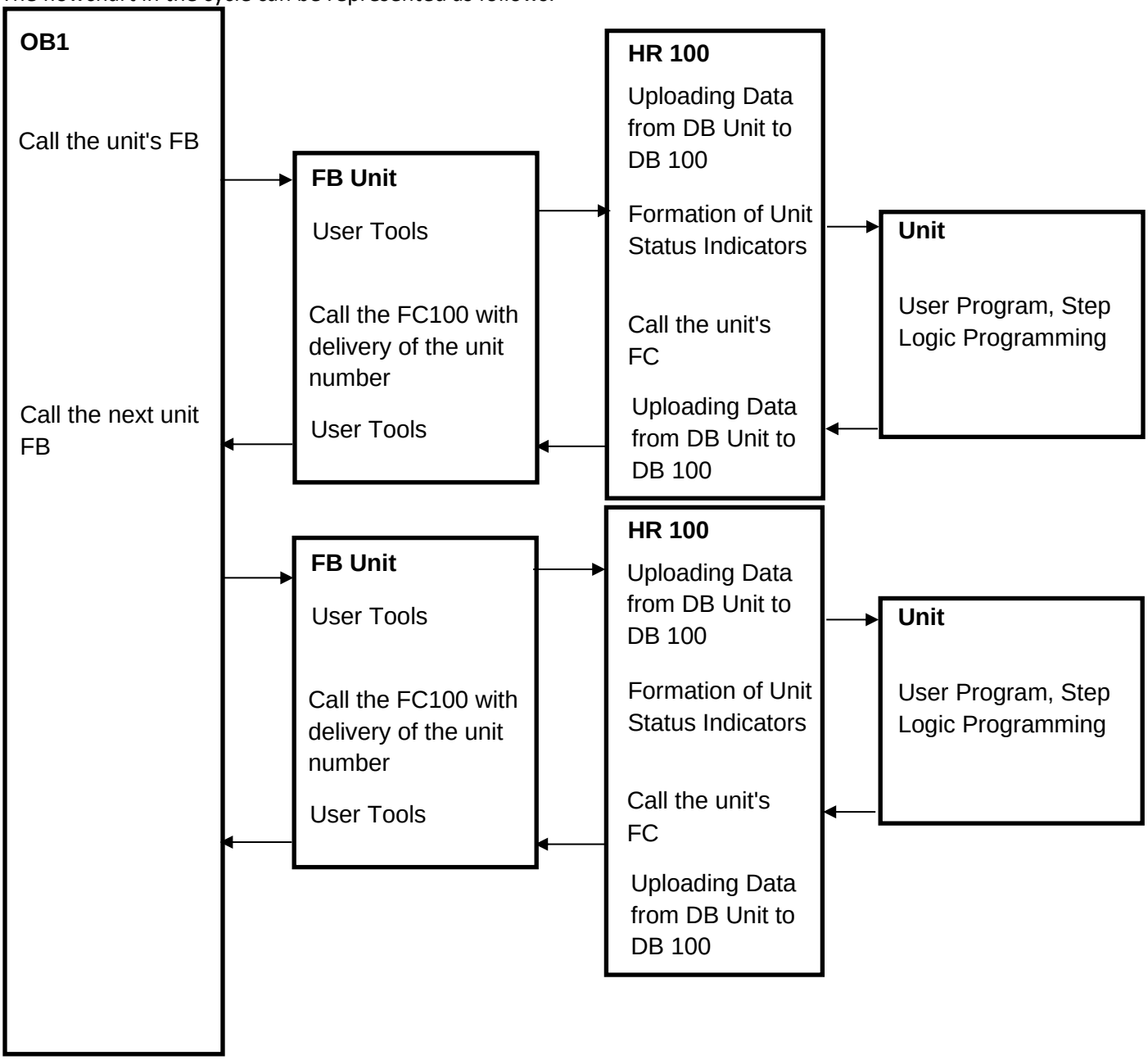
A "unit" is a production unit, such as the Fermentation tank, Pasteurizers or Filter, called in the above systems and sequencer. Programming the units is a real and central task. In order to achieve quick and easy programming, provided by a large number of tools.

Basically, a unit is always a DB, a FB, and an FC that is permanently assigned. Due dates and recipe values are stored in DB. In FB, fundamental parameter assignments are completed and the block function of the Unit-master (FC 100) is called. In FC you can program the steps (activities, actions).

The distribution is as follows:

- Unit 1 DB 101, FB 101, FC 101
- Unit 2 DB 102, FB 102, FC 102
- Unit 120 DB 220, FB 220, FC 220

The flowchart in the cycle can be represented as follows.



Basically, this diagram represents the execution that will be performed in a cycle:

- first starts with the OB1 where the unit's FB is called
- In the FB, actuator locks are performed, process values and parameters are transferred, and the FC of the same unit is called.
- In the FC, the executions that must be carried out in each step are carried out, counters are reset, etc.

15.1. One-step completion

When a step of a recipe is completed on a unit (PhaseEnd = TRUE), the following actions are performed in the same cycle without interruption.

- The current step is executed and detects the final condition = 1
- The current step is processed again with identifier "PLCycle" (last cycle)
- Entering the current data into the registry
- Recipe values are loaded
- Status indicators and parameter modules are updated
- New step is processed with identifier "PFCycle" (first cycle)

16. Unit Parameter Module

For the transfer of the nominal values of the BatchXpert recipe, or the recording of the flow values, the parameters are used. There are parameters from 1 to 40. A maximum of 16 parameters can be transferred per Step. It should be noted that parameter 1 is always used as a timer to monitor the passage. For the delivery of modules with nominal values and parameterization automatically from the standard program.

16.1. Value

BOOL	Enable Impulse Alarm	SHE
BOOL	enable low low alarm	EHHA
BOOL	enable high high alarm	xAl
BOOL	alarm from extern	ResetBlock
BOOL	Interlock counter reset	xSig
BOOL	Impulse Input	B30
BOOL	spare	B31
BOOL	spare	MLLA
BOOL	Low Low Alarm - Alarm if enabled	MLL
BOOL	Low Low Limit - warning if enabled	ML
BOOL	low limit	Msp
BOOL	setpoint	MH
BOOL	High Limit	MHH
BOOL	High High Limite - warning if enabled	MHHA
BOOL	High High Alarm - Alarm if enabled	ImpHM
BOOL	Impulse Help Memory	GAlQuitt
BOOL	General Alarm Quitt	Ign
BOOL	ignore alarm	Sim
BOOL	Simulation	Reset
BOOL	Reset counter	iELLA
BOOL	Counting Reserve	iEHHA
BOOL	enable HH alarm	iELLW
BOOL	enable LL warning	iEHHW
BOOL	enable HH warning	Gal
BOOL	General Alarm	Gals
BOOL	General Alarm Save	Warn
BOOL	General warning	Imp
BOOL	Impulse Flank	B04

16.2. Actual Parameter Modulus

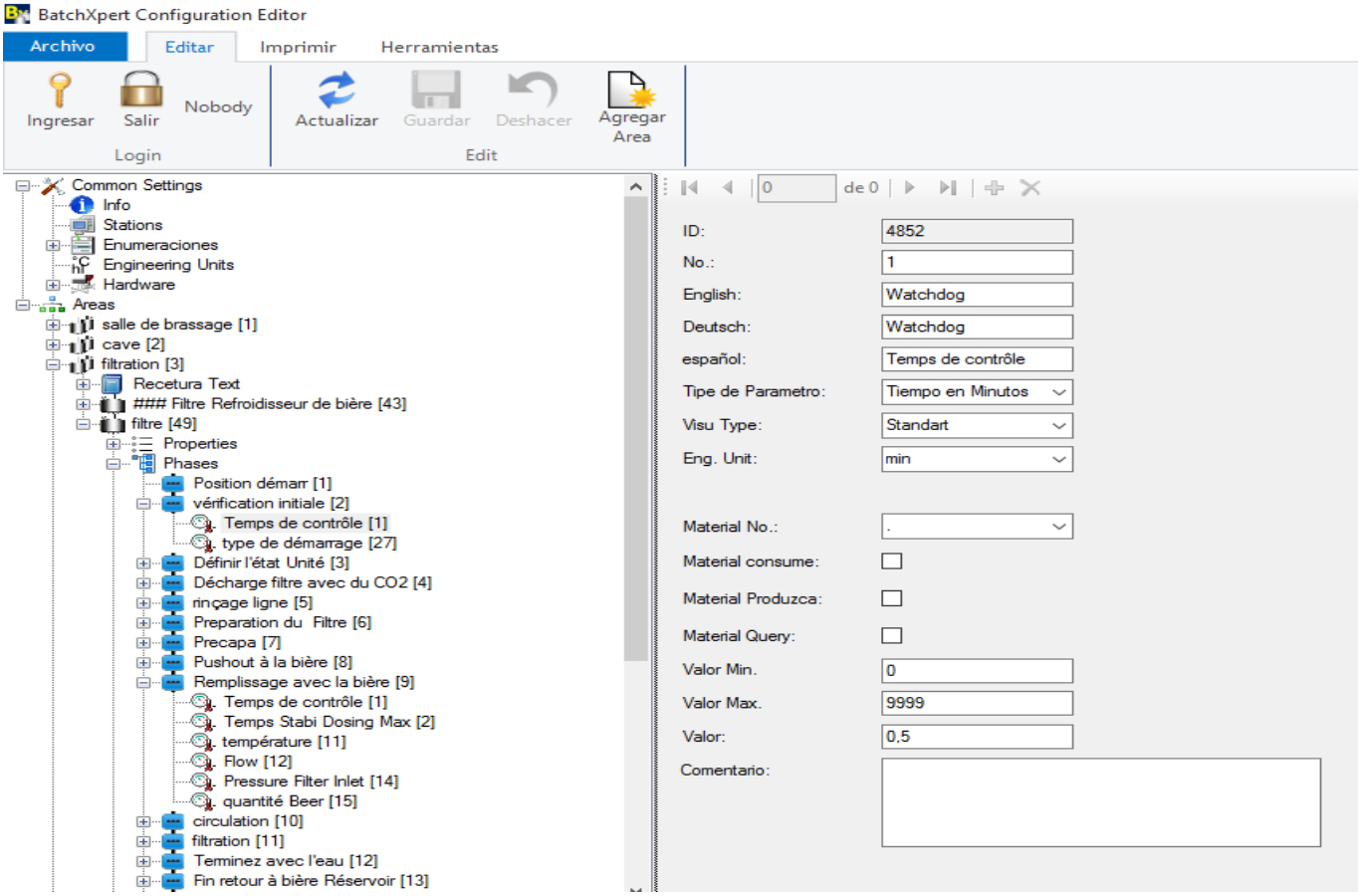
BOOL	spare	B05
BOOL	spare	B06 0 = valor de tiempo de la señal se pone a cero
BOOL	spare	User
BOOL	Memory Free for User	PVal
REAL	Process Value	Sp Esta variable es el valor real de maniobra Cuando se configura como un momento en que no se ha clasificado

16.3. Value OK

REAL	setpoint
LScal	REAL Valor real (Val)> = valor nominal (Sp)

16.4. Setpoint for Parameter Module

Low scaling	HScal
REAL	High Scaling Se otorga el número correspondiente, que también se utiliza en el programa
LLAVal	REAL
Low Low Alarm Value	LLVal
REAL	Low Low Value (Warning Limite) Para setpoint se utilizan textos de la pantalla Ejemplo: agitador OFF u ON
LVal	REAL
Low Value	HVal
REAL	High Value
HHVal	REAL
High High Value (Warning Limite)	HHAVal Sólo color en la pantalla de visualización
REAL	High High Alarm Value Es un código de color en la visualización El operador puede introducir manualmente el valor
ADVal	REAL Sólo color pantalla en la visualización
alarm delay value	Adsp



Example of a Parameter module configuration in the BatchConfigurator

16.5. Change Setpoint at Run Time

Parameter module setpoints can be overridden at runtime using the Unit Window. Of course, these types of actions can be detected, for the Protocol for Manual Interventions.

16.6. Programming Examples

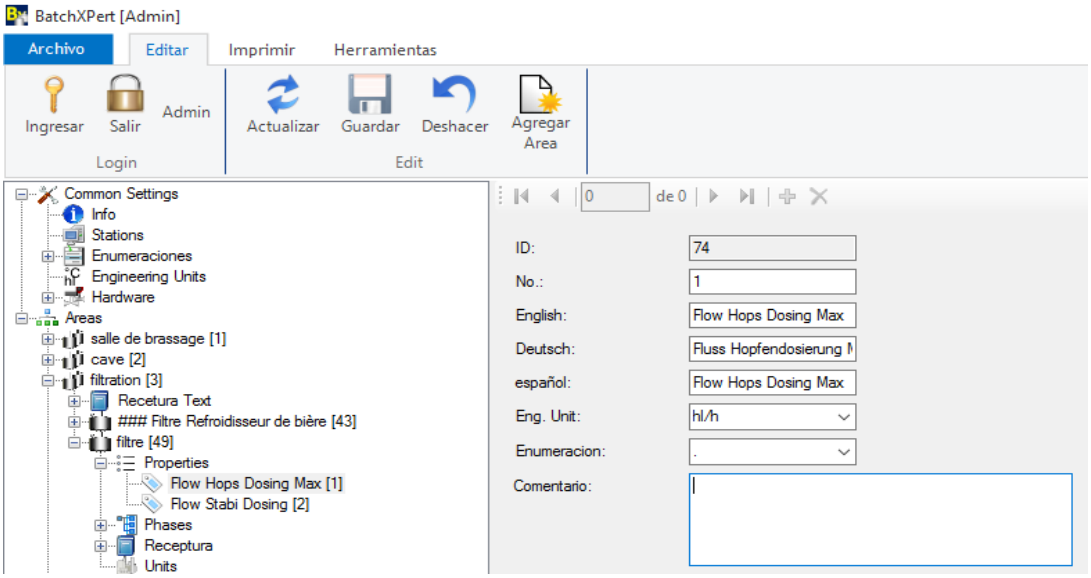
```
U "PA"
S "Uxx". To[2]. S          Launch the Parameter Module

A "DIn". DIn[12]. Gis      LSL
ON "Act". Act[32]. Out          Bomb
ON "PH"
S "Uxx". To[2]. H

U "Uxx". To[2]. D          Hit or exceed the setpoint of the parameter value.
 = "PhaseEnd"                   End of step
```

17. Unit Ownership

Configurations or properties of a special drive can be stored in the Drive's Properties.

17.1. Configuration (BatchConfigurator)

The configuration (ad text, unit, ...) is done at the class level in the configuration editor



17.2. Enter the values for each Unit

The variable with values is supplied via the Display Unit in the Properties window.

[1] volume max / hl:	1.00
[2] volume min / hl:	2.00
[3] volume release h.zone 2 / hl:	3.00
[4] automatic acid dosing:	Ein
	5.00
	6.00

17.3. Programming Examples

A unit property is associated with a unit to calculate the available quantity.

```
L "Uxx". Property[2]                    Tank Pipe Volume
T "UnitCom". U.Val1
```

18. Unit Status Indicators

The unit status indicators serve as an additional programming aid. These variables can be read if a unit is Hold, if the step has been run through the first cycle. It also informs about these variables if a fault is present, the process must be stopped or the step terminated.

18.1. New Step Number (Skip)

REAL	alarm delay setpoint	ImpVal
REAL	Value Per Impulse	CVal Señala que una falla está presente, se detiene el proceso de producción
DINT	counter value	Estado de unidad en alarma Señala que una falla está presente, se detiene el proceso de producción
Allocation	Default	Comment Se espera una acción por parte del operador, Señalizado en la visualización
EAImp	0	Trigger Pulse Monitoring Monitors if a pulse has ever been recorded If not, an alarm is triggered Se utiliza sólo para las tareas de registro especial. Cambio de paso o cambio de estado se escriben automáticamente los datos en el registro.
SHE	0	Low Low Alarm Activation When falling below the Low Low alarm limits, an alarm is triggered
EHHA	0	High High Alarm Activation When the high high alarm limit is exceeded, an alarm is triggered En el modo “RUN” el programa salta a la etapa siguiente
xAl	0	External Alarm If this signal 1 is triggered as an alarm Ver macas 1,0-1,7 (estéril, limpio, usado, ...)
ResetBlock	0	If this signal is set to 1, the counter cannot be reset

18.2. Current Program Number (Type Number, Type Identifier)

xSig	0	External Pulse Signal
User	x	Free for Programmer-Specific Applications Unidad es estéril
StatusClean	Allocation	Comment Unidad es limpiarla
MLLA	Alarm Limits Under	MLL La unidad no está limpia
Low Limit	ML	Low LimitPVal > LVal => 1-Signal A hysteresis is assigned to this value
Msp	Desired value OK	MHProducto 2 Unidad fue utilizada con el producto tipo 2
High Limit	MHH	High Limit HighPVal > HHVal => 1-Signal This value is assigned a hitheresis
MHHA	High Limit Alarm High	GalProducto 4 Unidad fue utilizada con el producto tipo 4

Alarm General Analog	Gals	Stores General Alarm General Alarm was or is activated
Warn	Operator Warning	Imp Unidad en el modo “RUN” (totalmente automático)
A	PVal	Process Value Unidad en modo de pausa (semiautomático) Aunque la unidad conduce al paso actual para llegar a la condición final, pero entra en modo HOLD. Permite que el operador adelante o retroceda pasos (paso +1 paso -1)
Hold	Allocation	Comment Unidad en modo HOLD (Detenido) El proceso de producción está en pausa (parada controlada) Permite que el operador adelante o retroceda pasos (paso +1 paso -1)
GAlQuitt	Recognizing an Alarm (Gals Reset)	 Ign El proceso de producción se detiene inmediatamente (parada no controlada) actualmente este modo no se implementa
Ignore Alarm	Sim	Simulation Mode The value "PVal" is manipulated directly in the display Posible sólo en el paso 0 es posible al inicio del programa, no en automático o manual La unidad queda bloqueada
Reset	Reset Counter	iELLA El estado de la unidad es incorrecto (watchdog, solicita Hold,...)
Triggers Low Limit Alarm	iEHHA	High High Limit Alarm Trigger : An alarm is generated if "PVal" is greater than "HHAVal" Señal debe restablecerse por el operador (OK + Reset alarma)
iELLW	Triggers Low Limit Warning An	iEHHW La unidad se encuentra en la Fase 2 (registro de entrada de inicio) se revisan la condición de inicio para el programa actual Si no se cumple la condición de arranque, no hay saltos de paso
High High Limit Warning Active	Sp	Face value Control de tiempo para el paso actual Automáticamente se activa la alarma (GAl)
LScal	Low Scaling	HScal La unidad está en el paso 0
Scaling High	LLAVal	Low Low Limit Alarm An alarm is generated if it falls from the low low limit Número de receta fue entregado y los datos de receta se ha leído con éxito de la base de datos
LLVal	Low Low Limit Warning	 LVal. La unidad no está en el paso 0
Low Limit	HVal	High Limit La Unidad se encuentra en el modo de CIP (Limpieza)
HHVal	Warning High High	HHAVal 1 = señal Paso activo Esta señal siempre es 1 en el paso actual
Alarm High High	Adsp	Alarm Delay: If the status is incorrect, an alarm is generated after xxx seconds.Paso activo y en el modo RUN, PAUSE o HOLD

ImpVal	Value of a pulse	Paso activo con función de retención 1 = señal de Paso activo y modo RUN o PAUSA Se utiliza para el control de los componentes activos de la planta
PP	M 5.3	Paso activo con función de retención 1 = Paso activa y modo de ejecución
Start	M 5.4	Botón de inicio de unidad El operador ha presionado el botón de arranque la unidad
OperatorOK	M 5.5	botón de unidad de operador OK El operador ha confirmado que fue realizada la acción manual requerida
PFCycle	M 5.6	Paso de primer ciclo El paso se acaba de activar. Primer ciclo
PLCycle	M 5.7	última fase del ciclo El paso ha alcanzado la condición de fin y está desactivado Último procesamiento del Ciclo
UnitNo	MB 10	Número de unidad actual
Phase	MB 11	Paso activado
StepNo	MB 12	Número del paso activo
Charge	Allocation	Guy Número de lote de producción
Comment	B24	BOOL ID número de producción único
Spare	B25	BOOL

18.3. Programming Example

Special Function in Primary Processing

```
U "PFCycle"
S "Cnt". Cnt[2]. Reset          Counter Reset
```

Actuator Control

```
U "PH"
S "Act". Act[12]. Aco           Activates the actuator in automatic mode
S "Act". Act[13]. Aco
S "Act". Act[14]. Aco
S "Act". Act[18]. Aco
```

End the step with the

```
U "DIn". DIn[17]. Gis           LSL
Or "Uxx". To[3]. OK             Maximum Time
= "PhaseEnd"                    End of step
```

Pass-through with operator request

```
U "PH"
S "OpReq"

U "OperatorOK"
S "PhaseEnd"
```

Arrest Request

U "Act". Act[12]. Gals
Or "Cnt". Cnt[2]. Gals
Or "Watchdog"
S "HoldReq"

19. Drive Boot Options

The initial supply of variables is done by the programmer (code system, the production planning system). Starting with production then, these variables are passed from one unit to another.

19.1. Start Option Value

Spare	B26	BOOL
Spare	B27	BOOL
Spare	xAlarm	BOOL
signal extern for alarm condition	B29	BOOL
Spare	B30	BOOL
Spare	B31	BOOL
Spare	B16	BOOL
Spare	B17	BOOL
Spare	B18	BOOL
Spare	B19	BOOL
Spare	B20	BOOL
Spare	B21	BOOL
Spare	B22	BOOL
Spare	B23	BOOL
Spare	GAlQuitt	BOOL
General Alarm Quitt	Ign	BOOL
ignore alarm	Sim	BOOL
Simulation	OPMsg	BOOL
operator message	B12	BOOL
spare	B13	BOOL
spare	B14	BOOL
spare	B15	BOOL
spare	Gal	BOOL
General Alarm	Gals	BOOL
General Alarm Save	OPMsgActive	BOOL
Operator Message Active	AlarmMsgActive	BOOL
alarm message active	iAlarm	BOOL
alarm active intern	B05	BOOL
spare	B06	BOOL
spare	User	BOOL
free for user	ADVal	REAL
alarm delay current value	Adsp	REAL
alarm delay setpoint	BOOL	start option bit
Allocation	Default	Comment
xAlarm	0	Activation of the Message (Message)
User	x	Free for Programmer-Specific Applications
Val4	Allocation	Comment
Gal	Alarm General Analog	Gals
Stores General Alarm	OPMsgActive	Turn on Operator Message
AlarmMsgActive	Alarm message active	start option value

19.2. Programming Example

```
U "Uxx". StartOption.b01
= #WeakWort                    option 1 = with weak word

U "Uxx". StartOption.b02
= #Trub                        option 2 = With trub dosage
```

20. User Data

This area is reserved for user programming and serves as a clipboard for certain process data or memory for certain production processes. A declaration (documentation) of the variables used should always be stored as a block comment in network 1 of the pass module.

20.1. User Value

Allocation	Comment	GAlQuitt
Recognizing an Alarm (Gals Reset)	Ign	Ignore Alarm
Sim	Simulation Mode	OPMsg
Operator Message	Adsp	Alarm Delay (message)
b27	BOOL	user bit
b28	BOOL	user bit
b29	BOOL	user bit
b30	BOOL	user bit
b31	BOOL	user bit
b16	BOOL	user bit
b17	BOOL	Allocation
Guy	Comment	Set
BOOL	Switch Software Set	Reset
BOOL	Reset Software Switch	B26
BOOL	spare	B27
BOOL	spare	B28
BOOL	spare	B29
BOOL	spare	B30
BOOL	spare	B31
BOOL	spare	B16
BOOL	spare	B17
BOOL	spare	B18
BOOL	spare	B19
BOOL	spare	B20
BOOL	spare	B21
BOOL	spare	B22
BOOL	spare	B23
BOOL	spare	B08
BOOL	spare	B09
BOOL	spare	B10
BOOL	spare	B11
BOOL	spare	B12
BOOL	spare	B13
BOOL	spare	B14
BOOL	spare	B15
BOOL	spare	B00
BOOL	spare	B01
BOOL	spare	B02
BOOL	spare	Gis
BOOL	spare	B04

BOOL	spare	B05
BOOL	spare	B06
BOOL	spare	User
BOOL	free for user	user long int
Allocation	Default	Comment
Set	0	While
Reset	0	It serves as a switch to back up in the sequence program
User	x	Free for Programmer-Specific Applications
Val0	Allocation	Comment
Gis	Condition Software Switch Status	user value
Val2	REAL	user value
Val3	REAL	user value
Val4	REAL	user value
Val5	REAL	user value
Val6	REAL	user value
Val7	REAL	user value
Val8	REAL	user value
Val9	REAL	user value
Val10	REAL	user value
Val11	REAL	user value
Val12	REAL	user value
Val13	REAL	user value
Val14	REAL	user value
Val15	REAL	user value

20.2. Special Registration Programming Example

By using User Variables, the transfer of the current silos has a one-cycle delay. Thus, to activate a log entry in a silo it can be swapped, thereby logging the deletion of the previous silo.

Docu

UserDint 01: Memory of the next silo number

```
L "Dxx". User.Dint1DIn1
T "Uxx". For[13]. Val

L "SiloNo"
L "Dxx". User.Dint1
<> D
U(
L 0
>D
)
S "ProtWrite"

L "SiloNo"
T "Dxx". User.Dint1
```

21. Starting and Stopping a Run/Hold

In certain cases, the process should not be stopped abruptly or not all will be triggered immediately at the start. The reasons for this can be both economic, as well as avoiding current elevation technologically.
In order to achieve an orderly start and shutdown, there are two possibilities. In simple contexts, a delay time to turn on or off can be stored directly on the actuators, without the need for programming. In more complex conditions, a simple realization is achieved with the variables of units of time.

21.1. Step Time in "RUN"

Símbolo	Operando	Comentario
THold	REAL	unidad de tiempo en mantener Tiempo en segundos, tiempo de la unidad en espera
TRun	REAL	unidad de tiempo en “RUN” Tiempo, en segundos, tiempo de la unidad en funcionamiento
TStepRun	REAL	Tiempo en “RUN” del paso Tiempo en segundos, tiempo que el paso actual esta en “RUN”

21.2. Programming Example

The following example shows a simple orderly shutdown. The status of the unit is on HOLD, either by an internal booking request event of a malfunction, or the operator switches the unit to HOLD, so the THold time starts working. For example, actuators 12 and 13 are immediately stopped, delayed from the actuator 18 to 10 seconds and delayed the actuator 19 to 15 seconds.

```
U "PH"
S "Act". Act[12]. Aco
S "Act". Act[13]. Aco

L "Uxx". U.THold
L 1.000000e+001                 10 seconds
<R
S "Act". Act[18]. Aco       Delay On

L "Uxx". U.THold
L 1.500000e+001                 15 seconds
<R
S "Act". Act[19]. Aco       Delay On
```

22. Unit-to-Unit Communication

One of the most elaborate jobs is to synchronize the units with each other. To accomplish this task quickly, there is a standardized interface provided by the BatchXpert. Each unit has the ability to communicate simultaneously with 4 Masters and 4 Slaves. It is destined in 4 masters or 4 slaves of different classes.
Example:
The exchange of wort cooling data to 99 fermentation tanks occupied a single level. Therefore, the cooling of the wort has as "slave 1" a fermentation tank. Slave compound 2 can be a yeast tank or propagator, etc.

22.1. Name Specification

- UnitCom.U.Master1.xxx Current Unit Signals in Master 1
- UnitCom.Master1.xxx Master 1 Signals in the Current Unit
- UnitCom.U.Slave1.xxx Signs of the current unit on the Slave 1
- UnitCom.Slave1.xxx Slave 1 signals on the current unit

For Master 2 to 4, or slave 2 to 4, the same schematic applies

22.2. nominal value or the actual value of the associated unit

Asignación	Default	Comentario
UnitCom.U.Master1.TransReq	0	Solicitud de Transferencia
UnitCom.U.Master1.TransActive	0	Transferencia Activa
UnitCom.U.Master1.TransEnd	0	Transferencia Terminada
UnitCom.U.Master1.TransRel	0	Liberar control de Trasnferencia
UnitCom.U.Master1.FillReq	0	Llenado solicitado
UnitCom.U.Master1.FillActive	0	Llenando activo
UnitCom.U.Master1.FillEnd	0	Llenado terminado
UnitCom.U.Master1.FillRel	0	Libera control de llenado
UnitCom.U.Master1.FunctionNo	0	número de función especial Para sincronizaciones de flujo complejo
UnitCom.U.Slave1.TransReq	0	Solicitud de Transferencia
UnitCom.U.Slave1.TransActive	Allocation	Guy
Comment	S	BOOL
Start Parameter Modul	H	BOOL
Hold Parameter Modul	Reset	BOOL
Reset Parameter Modul	OK	BOOL
OK	S04	BOOL
spare	S05	BOOL
spare	S06	BOOL
spare	S07	BOOL
spare	No	BYTE
ParamodulNo	OnlySp	BOOL Para sincronizaciones de flujo complejo
only setpoint	OnlyVal	BOOL Para maniobrar el número de unidad de Asociado
only value	Enum	BOOL
enumeration	TSec	BOOL
Time in sec	TMin	BOOL
time in minutes	THour	BOOL

time in hours	TDay	BOOL
time in days	S27	BOOL
spare	Endcond	BOOL
Phase End Condition	ManuInput	BOOL

22.3. Value 8

Manual Input Required	AlarmCond
BOOL	alarm condition
S33	BOOL
spare	S34
BOOL	spare
S35	BOOL
spare	S36
BOOL	spare
S37	BOOL
spare	Sp
REAL	setpoint
Val	REAL
Value	ignorar la alarma
Allocation	Default
Comment	S
0	It will only be used if the parameter module is set as a time signal, 1 = Time started
H	0
It will only be used if the parameter module is set as a time signal, 1 = Time stops	Reset
0	Parameter Module Reset
Val	x
Actual Parameter Modulus	Almacena Alarma General
UnitCom.Master1.SCE	Allocation
Comment	OK
Value OK	unidad en el paso 0
UnitCom.Master1.ReadyStart	Allocation
Comment	No
Module Number Parameter	OnlySp
It is used only as a nominal point of the parameter module	OnlyVal
Used only as an actual value of the parameter module	Enum
Enumeration	TSec
Setting the Time Module on a Second Base	TMin
Configuring the Time Module on a Minute Basis	THour

Configuring the Time Module on an Hourly Basis	TDay
Setting the Time Module on a Day Base	Endcond
Final Condition of the Color Only Parameter Module	ManuInput
Manual Input to the Parameter Module	AlarmCond
Alarm Condition of a Parameter Module (Alarm Limit)	Sp
Setpoint for Parameter Module	PrId cambio del asociado La unidad asociada se inicia con un nuevo PrId Por ejemplo, en un cambio de rango Sin embargo, siempre es recomendable utilizar un código de sistema especialmente adaptado a un cambio de rango
UnitCom.Master1.FunctionNo	Número de función especial Para la sincronización de flujo complejo
UnitCom.Master1.UnitNo	Número de unidad asociada
UnitCom.Master1.PrId	PrId
UnitCom.Master1.Charge	Número de lote Número Producción
UnitCom.Master1.ProgNo	número de programa Normalmente escrita en la identificación del producto
UnitCom.Master1.Val1	valor 1
UnitCom.Master1.Val2	Valor 2
UnitCom.Master1.Val3	valor 3
UnitCom.Master1.Val4	valor 4
UnitCom.Master1.Val5	valor 5
UnitCom.Master1.Val6	valor 6
UnitCom.Master1.Val7	Valor 7
UnitCom.Master1.Val8	valor 8

Additional signals and values to masters and slaves must match schematic

22.4. Programming Examples

Connect with Slave

Communication can be established to a slave, if the slave's unit number is transferred, and the "Connect" signal can be established. For most compounds, the slave's unit number can be transferred statically, as this does not change.

Example of a straight production line or with a brewery brewing line. The slave unit number can also be dynamically assigned. According to certain criteria of the program, wort boiler 1 or 2 of such dynamic selection can be used together, however, they can be taken by a code system (Production Planning System, Production Calendar System).

```
A "Step0"
S "UnitCom". U.Salve1.Connect
L 13                                    Associated Unit Number = 13
T "UnitCom".U.Salve1.No
```

One-step synchronization flow

```
U "PA"
S "UnitCom". U.Salve1.Start
S "UnitCom". U.Salve1.TransReq

U "UnitCom". Salve1.FillActive
U "UnitCom". Salve1.Run
= "PhaseEnd"
```

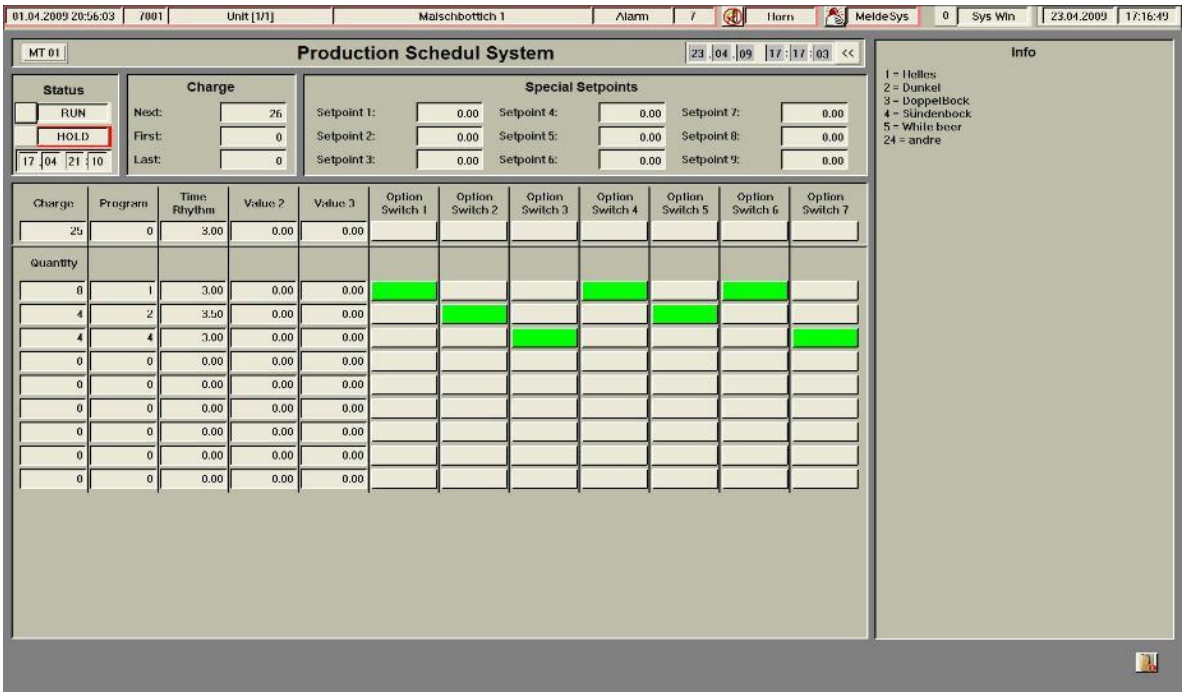
Release for Activations

When controlling the active components for the transfer release should always be consulted (Fill or Release). Therefore, a mutual shutdown in the event of a breakdown or if an operator sets the Unit Retention very easily.

```
U "PA"
S "UnitCom". U.Salve1.TransActive
U "PH"
S "UnitCom". U.Salve1.TransRel


U "UnitCom". Salve1.FillRel
U "PH"
S "Act". Act[48]. Aco
S "Act". Act[49]. Aco
```

23. Production Planning System

It is a simple code integrated into the production system. Here you will find the production plan, number of productions, program number (specification type), production rhythm and some Start options. How to use the boot options is up to the programmer, the system in "Run" mode, batches are processed one after the other. The start of the respective Unit takes place once it is free. With the start time (see Status) you can switch from Hold state to Run mode automatically.
The planned time for the production rate is passed on a deferred batch.



Startup options are automatically transferred:

- Production System Bits -> Start Option Bits
- Production System Production Rate -> Start Option Value 1
- Production System Value 2 -> Start Option Value 2
- Production System Value 3 -> Start Option Value 3

Special setpoints have the ability to add setponit for one step. For example, a malt compensation during the first brew, or an offset for the volumes of the last brew. The evaluation of the first or last production is carried out through the batch values. First, load the batch.

23.1. Programming Example

To use the production system, call FC 281 and transfer the corresponding unit number. It can be used as a starting unit, of course, several units. In this case, the unit number must be passed in the form of a variable, which is the number of the corresponding unit. Don't call the block repeatedly.

```
CALL "xProdSchedul1"
 UnitNo:=1
```

24. "Visu Extern" System (Touch Screens)

The BatchXpert system implements a sophisticated concept of data exchange between the controller and the HMI systems. The system allows you to aggregate all types of HMI systems, and process the data to these HMIs independently for each HMI. This system is useful when there are several HMI systems connected to the system that contain mechanisms that operate differently from each other. By this he means the operation of the SCADA system, running on the HMIs, and not the process operation of these systems.
For example, it is possible to connect a Siemens HMI in parallel with text displays or exotic systems that require specialized data exchange.

Generally, the BatchXpert system separates between two types of Visualization (HMI)

- **Internal View:** The main system, integrated into the BatchXpert System. Only the "GraphPic" SCADA system from the company "Gefasoft" is available
- **External View**: All other HMIs. For example, Siemens WinCC Flexible, Progea Movicon, or other types of touch screens.

Regardless of the type of Visualization (Internal or External), the system manages and records all the actions performed to some control module, in such a way that there are records in the system's databases. This means that there are also Records on Manual Operations not only of the Internal View, but also of all the External Views added to the system.

The flow of commands and statuses that are displayed/activated in the connected HMI systems follow the following conceptual flow.