BATCHXPERT

Manual para Programadores

Español V1.1

Content

1. Preface

The BatchXpert system is a Control system and visualization of processes throughout complete production plants, incorporates reporting protocols, Batch Recipes, Batch trackin with Material handling, trends, alarms on BatchXpert Stations or Panel based HMI, etc...
For more simple applications there Compact BatchXpert system this system is for processes that do not require reports protocols. batches or recipes, but incorporates historical trends and alarms and HMI, etc ...

The PLC program is implemented on the basis of Simatic S7, so it can be used in all Simatic S7 compatible PLCs, such as Speed7 from Vipa or S7 from Siemens. The PLC program of BatchXpert is completely standardized with modularity and ease of use for the programmer in mind. It includes standardized program blocks, Interfaces and control modules, to significantly reduce Engineering times of a BatchXpert application. This PLC program is design with various characteristics in mind - from short cycle times to maximum of standardization of both hardware and software. Of course the Application implements various configurations, so that the standardized components can be adjusted to the project's needs.

Some features of the BatchXpert system are:

- 120 units (sequences)
- 1600 actuators (motors, valves, ...)
- 1600 Digital Input (Trigger, Vacuum sensors, ...)
- 480 analog inputs (temperatures, pressures, ..)
- 320 Measurements (Flow Measurements)
- 320 Regulators (PIDs), and several more modules ..

2. General information about the HMI

2.1.Screen resolutions:

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

- Display **GraphPic**
  - 1600 X 900 (HD)
  - 1920 x 1080 (Full HD)
  - 1366 x 768 (Notebook)

- Display **Movicon** for touch screens
  - 1024 X 768 (10 ')
  - 800 x 480 (7 '')

- Display **WinCC Flexible** for touch screens (without TIA Portal)
  - 800 x 480

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

2.2.Adjust screen resolution

GraphPic

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

Movicon

- There is a library of the Movicon "BatchXpert SDK" / Visu / Movicon Vx.x.x.x (unzip into the folder of the project).
- "Select the folder with the resolution that need, example: for a 7 '' screen keep the folder"Dev 800 x 480"(this folder be must be renamed with the name of the project)
- Remove remaining folders (folders that do not use library)

Note: all resolutions are modifiable. However this usually requires several considerations annexes, so consult Mlogics a problem as well.

2.3.Host names

In a BatchXpert system, the default operation stations are adjusted in the following names, which are adjustable, but you should keep the standard system.

- BX01 for the first server
- BX02 for the second server
- BX03 for the third serve theor
- BXSlave1 for the first slave
- ...

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

1. **BX.Ini**: for the name which is located in: "**C:/Daten/BX.ini"** under the settings key **"StationName ="**
2. **HostName**: If there is no name in "StationName" assumes the HostName (computer name)
3. **Slave**: If you don't have a name assigned to a "0" at the end of the name

Note: An Programmer should have the BatchXpert station set **as a slave.** This is to avoid conflicts with if other stations ar set to the same 'master' in the system.
To assign a “Slave” as BatchXpert master number, go to "BatchXpert Management Console" at "BatchXpert Station" changed to "Slave".

2.4.Directories important

BatchXpert depends on some folders important to be able to function properly. In:
C:\Daten
There are variouss folders that correspond to the various functions of the system.

- **Backup**: withhave backups of the system
- **Engineering:** It contains detos of engineering such as imports give them taglist
- **Logs**: contains information on lyou every tools event

Programs (x 86) C:\archivos \BatchXpert
There are folders where you can find information and also the executable tools of BatchXpert.

- **Documentation:** It contains manuals for programmers and operators in different languages on the use of the BatchXpert, also the BatchXpert license information.
- **Reports:** in this folder is convervan the templates of different reports that can be generated BatchXpert (Batch Sumary, Recipe thesaurus, thesaurus With Menu...).
- **Tools:** are additional tools of the BatchXpert)Comandline Tools).

Programs (x 86) C:\archivos \BatchXpert SDK
There are additional tools of the BatchXpert and planpads for the programmer, This focused on the compatibility of the different softwares with the BatchXpert system and HMI, to minimize the tiEmpo of engineering, requires the **installer of the BatchXpert SDK**.

- **Documentation:** It contains manuals for programmers on the use of the tools of the BatchXpert, HMI and PLC.
- **PLC:** contains templates for the S7 PLC (compatible with Vipa)
- **Tools:** It contains templates for the documentation of the project (tagliste, Project Information) and executable tools)Installation Center, Comandline Tools).
- **Visu:** There are templates for displays (GraphPic, Movicon, WinCC Flexible).

Programs (x 86) C:\archivos \BatchXpert Micro SDK
There are templates for the programmer, is designed for PLC reduced memory, compatibility with the HMI and minimize the engineering time, requires the **installer of the BatchXpert Micro SDK.**

- **Documentation:** It contains manuals for programmers on the use of the HMI and requirements of the BatchXpert Micro.
- **PLC:** contains templates for PLC S7-300, S7-1200 and RSLogix5000.
- **Visu:** There are templates for displays (Movicon, Factory Talk View).

3. Generalidades about the PLC

The BatchXpert consists of a program system Basic, or "Operating system" in a certain way, abstrahe user access and direct handling of the periphery program, providing additional advanced features such as timers, status, control modules,...

The program of the PLC usually is structured in the following way:

- FC 1-100: System functions fixed (not can also be reallocated block numbers
- Db1-100: Data from the system fixed, without possibility to reassign.

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

Fixed the FC of the BatchXpert system system, can appear as a very rigid for a programmer, but actually this Convention facilitates programming both the system as the user's project also.

3.1.Requirements to the PLC

The BatchXpert system requires certain features of the PLC. The system requires above all lots of RAM to funionar. For more information, please review the "System Requirements" manual of the system.

The program is compatibe with lines Siemens **S7-300, S7-400** and the **VIPA Speed7**. The PLC Simatic **S7-1200** This only compatible with the system "Micro-BatchXpert", which is similar to the BatchXpert systems, but form an own system, with less range than the "large" systems.

3.2.Important global signals

The BatchXpert system provides the following signals for use in the user program. All the signals presented are "Read only" and should not be written by the user. The following signals are comprehensive, and can be used throughout the program.

General status of the PLC

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	878.7 M	BOOL	reset all alarms
PLCNo	MW 998	INT	Number of PLC within the BatchXpert system

Timers and cycles

Clk2CE	M 879.0	BOOL	clock cycle 2 (edge)
Clk4CE	M 879.1	BOOL	clock 4 cycle (edge)
Clk8CE	M 879.2	BOOL	clock cycle 8 (edge)
Clk16CE	879.3 M	BOOL	clock 16 cycle (edge)
Clk32CE	M 879.4	BOOL	clock cycle 32 (edge)
Clk64CE	M 879.5	BOOL	clock 64 cycle (edge)
Clk128CE	M 879.6	BOOL	clock cycle 128 (edge)

Clk256CE	M 879.7	BOOL	clock cycle 256 (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	881.2 M	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)
1 sec	M 895.0	BOOL	tact 1 sec
2 sec	M 895.1	BOOL	tact 2 sec
4 sec	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	M 895.6	BOOL	64 sec tact
128sec	M 895.7	BOOL	68 sec tact
CycleCnt	MB 879	BYTE	cycle counter
CycleTimeMin	MD 904	REAL	time minutes
CycleTimeHour	MD 908	REAL	time hours
CycleTimeDay	MD 912	REAL	time days
CycleTimeSec	MD 900	REAL	Cycle Time in Seconds
TimeMSecInt	MD 916	DINT	time in milli seconds in DINT

3.3.Creation of program with the help of the HMI

Assuming the controller programming is performed after you have created the graphics of the HMI, graphic used can beTsar as an aid. With a click en the object the number, which also is visiblee is used in programming, simple numbers are more faciles keep short term that a complex code Tags.

Eork various units for a class, so it is advisable to create a DB class.

1. All objects are classifican the DB
2. PRogramming from stages, Evaluation, Locks, ... about classes of DB
3. Objects of class in the new DB class

3.4.Program structure

In the following shows the structure of the General system calls.

The functions are colorizados according to the following categories:

- The system functions, non-modifiable
- Functions related to the IO, adjustable if required
- Functions related to the sequcias of the process (unit), adjustable by the user

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)	

Usually the user programs are programmed in the FB1xx and the FC1xx of the seqcuencias. The IO-related blocks are generated by the tool of the BatchXpert system engineering.

4. Difference between the BatchXpert and BatchXpert Compact

The two systems, BatchXpert and BatchXpert compact are very similar, and much the user code can be kept unchanged the PLC. However the BatchXpert system Compact is designed for projects more small without any reports.
"BatchXpert Compact" system includes the modules "BatchXpert" system control and has the same functions and control standards The system is perfect for controlling processes and machines that do not require advanced systems of registration of historical data and process functions as there are in the "BatchXpert" System.
"BatchXpert Compact" system includes basic historical trends data and records of alarms systems.

Basically the difference between "BatchXpert" and "BatchXpert Compact" is that "BatchXpert Compact" system does not incorporate the functions of advanced historical records and the advanced features of process management. These logs are stored directly in the HMI screen or the computer of the SCADA system to make them accessible for analysis.

	BATCHXPERT	BATCHXPERT COMPACT
Database	Microsoft SQL2005 or more new.	There is no database. All data are in the PLC.
Historical data	Trends, alarms, logging of processes, manual operations, processes reports.	Trends and alarms.
Historical data redundancy	Up to 8 fully independent servers.	There are no records of historical data except trends and alarms.
Handling of sequences	Handling advanced programs stored in the database. Editable with the system configuration editor.	Basic programs stored in the PLC operation.
Connection PLC´s simultaneous	Unlimited.	A PLC for each touch screen.
Control modules	All.	All.
Recipes and programs	Unlimited and saved in the database.	Stored in the PLC and restricted to the memory of the PLC.
Limit HMI	There is no limit (allows up to 8 servers, but a) number unlimited clients).	There is no limit amount of HMI.

5. BatchXpert SDK and BatchXpert Engineering Tool

To facilitate the engineering and implementation of an automated with BatchXpert project, there is the "BatchXpert Software development Kit" also called "SDK". This package installs all tools of engineering and templates of the conroladores as well as display systems also.

The version most recent SDK, is available from the link siguetne:
http://www.mlogics-automation.com/?q=es/content/descargas-para-clientes

We recommend using the same version of the SDK that the version of the BatchXpert that is used in the plant. For new projects, it is recommended to use versions more recent, to take advantage of the support of the system.

Once installed, are the following folders within the installation folder:

- **Visu**: This folder contains templates of projects for all supported display systems of the BatchXpert system. You must copy the file appropriate to the engineering folder, detached and adjusted to accommodate the needs of the project.
- **PLC**: Contains templates of the PLC supported BatchXpert system . Just as with the visu, must be copy it to the engineering folder, remove and adjust the.
- **Tools**: Contains tools of engineering equipment, such as templates for Taglist, an Hermienta of SQL database management and several hermaientas more
- **Documentation**: It contains many manuals, that aren't included in the default installation of the BatchXpert, since are destined for a proposal engineer and not towards the operators.

"Also, several tools were installed as the"BatchXpert Project Engineering Tool"that lets you import, export and generate data for PLC and HMI on the basis of the engineering database current system. This tool can generate alarm messages to the HMI, blocks of data to the PLC and other more.

6. General structure of the DB of control modules

In the PLC, the object data (actuator, PID,...) are maintained in Arrangements, can also use DBs with di itvision of different elements. It is only important to keep the internal structure of the objects. Of course, the use of the various elements has the advantage on each item you can write the name of the tag and a comment. However, this only has sense a code tag can be used throughout the project. If the code tag is specified by the client, so you know the names tag of the elements, usually only if you have the project is almost completed.
Is prefierencia the division in Arrangements. To the create a project can be in the following way:

1. Enter the number of objects in the P & ID -Diagram
2. Create graphic images
3. Reading Excel tables with the name of the object
4. Programming of the S7

Templates of the PLC are all the same structure for programming, whether a template BatchXpert SDK or Micro SDK (SDK Micro not t)lene sequence, units, etc.)

6.1.Data structure

The structure of data is related directly to the DB and there is a block of data for each item in control (actuators, PID, etc.). These data blocks contain a long array, where each item corresponds to the module control with the number correspondiene and follows the structure described in this section.

6.2.Commands

This section describes the data that function as commands from the user to the BatchXpert system program. The signals described in this section can be written in the user program with the corresponding to each signal restrictions.
Estes generalmenet signals it's commands that activate the corresponding function in the control modules... Generally (with some exceptions), it is senaels of "Write only" or "Single writing".

6.3.Status

They are data and signals provided by the BatchXpert system for use in the user program.
It's status, where the user can obtain different control modules General status information. Generally (with some exceptions), it is senaels of "Read only" or "Read-only".

6.4.ParametrI

It's different conrol modules configurations. This configuration is normally done through the "Faceplate" of the HMI sistmemas of the BatchXpert. Normalmene are not written from the PLC.
However for some signs there is the possibility to manipulate the paraMeasurements from the PLC to enforce a charactaristica cirta, without giving the possibility to the user for mainipular paraMeasurements. For example reset stop emergency simulations...

7. Actuators (Act)

The actuators are all the outputs of the PLC who are related or not with the process (valves, pumps, up to or led lamps).

An actuator has basically an exit and up to two FeedBack.

The actuator associated with physical output takes place in FC 501 “TransAct”. (Esample in "APartNumber Act”).

A PID display is presented as an actuator, to indicate the status of the PID actuator, without however is only a softaware actuator, usually has no exit physical.

7.1.Structure data

Assign the	Type	Comment
ACo	BOOL	automatic control
ExCo	BOOL	extern control
SCS	BOOL	status check start
xFBa1	BOOL	1 feedback
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 external feedback (0 FBa1 = OFF FBa2 = ON / 1 FBa1 = ON FBa2 = OFF)
FBa1Active	BOOL	feedback 1 active
FBa2Active	BOOL	Feedback 2 active
xAutoHM	BOOL	extern automatik old
GAlQuitt	BOOL	General alarm quitt
IGN	BOOL	ignore alarm
SIM	BOOL	Ssimulation
Auto	BOOL	automatic mode
MCo	BOOL	manual control
EmRel	BOOL	emergency release
InterlockGAl	BOOL	interlock by alarm
Maint	BOOL	Maintenence
EA.	BOOL	General alarm
GAlS	BOOL	General alarm save
SCE	BOOL	status check error
Mov	BOOL	actuator is moving for visu
On	BOOL	actuator is ON
Off	BOOL	actuator is OFF
Out	BOOL	Output
User	BOOL	free for user programm
TOnVal	REAL	turn on delay value
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 before restart interlock
SwCntVal	DINT	switch counter value
RunTimeVal	DINT	duty timer value (seconds)

7.2.Commands

Assignment	Default	Comment
ACo	0	Directed to actuators Active only in automatic mode
ExCo	0	The external control, for example by a switch (digital input) Effective in automatic and manual Related interlocks with the production are bridged
SCS	0	Start health check consultation If the actuator is not turned off or altered as a check of the State error is indicated
xFBa1	x	FeedBack 1 It is the position of the actuator feedback
xFBa2	x	FeedBack 2 It is the position of the actuator feedback
Rel	1	Security lock They are vital safety things like emergency, door man stops.
Rel2	1	Production blocking Blocking of processes not critical but importanttant as level empty, high level.
xAuto	0	Signal Automatic (usually "RUN" that accompanies the) Unit)
User	x	Free for specific applications of the programmer

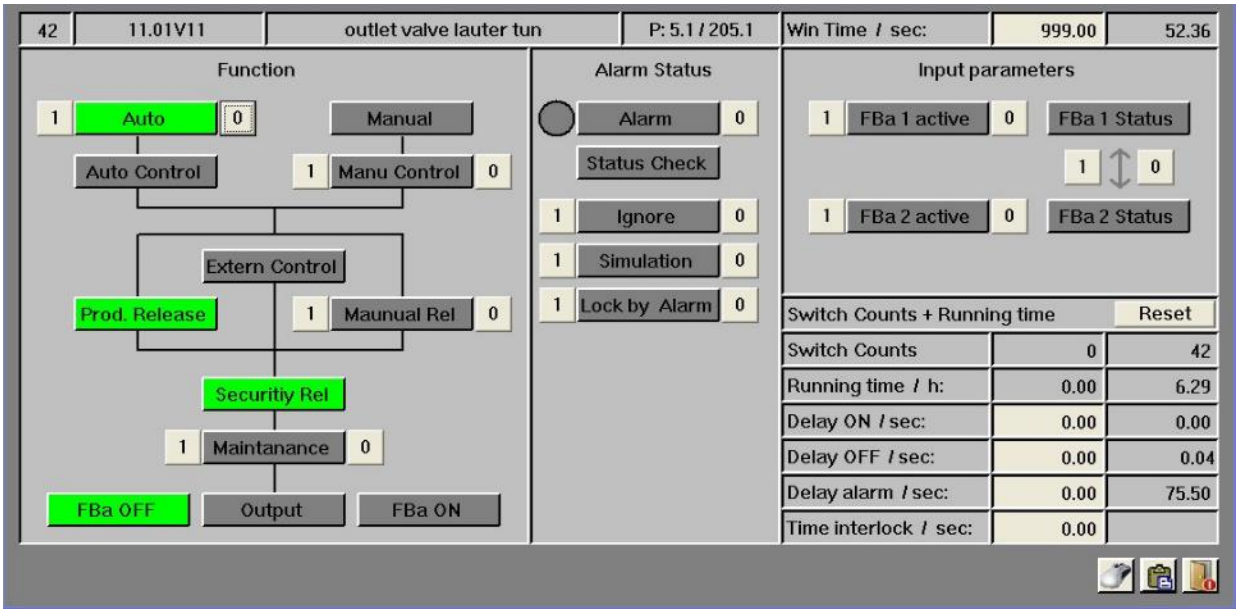
7.3.Status

Assignment	Comment
FBaOn	Feedback On of the actuator)Energized) ESTfeedback can be activated or in simulation, this signal is generated Internamete.
FBaOff	Feedback Off actuator (not energized) This feedback can be disabled or in simulation, this signal is generated Internamete.
EA.	General alarm States of actuators not Ok
GAlS	Stores General alarm General alarm was or is activated Signal must be reset por operator (OK + Reset) alarma)
SCE	Designates an error condition State incorrect actuators for the next program start
Mov	The actuator is by move Feedback still incorrect.
On	Actuador On (output and Feedback fired and without off feedback)
Off	Off actuator (no output or feedback and feedback off)
Out	Output

7.4.ParaMeasurements

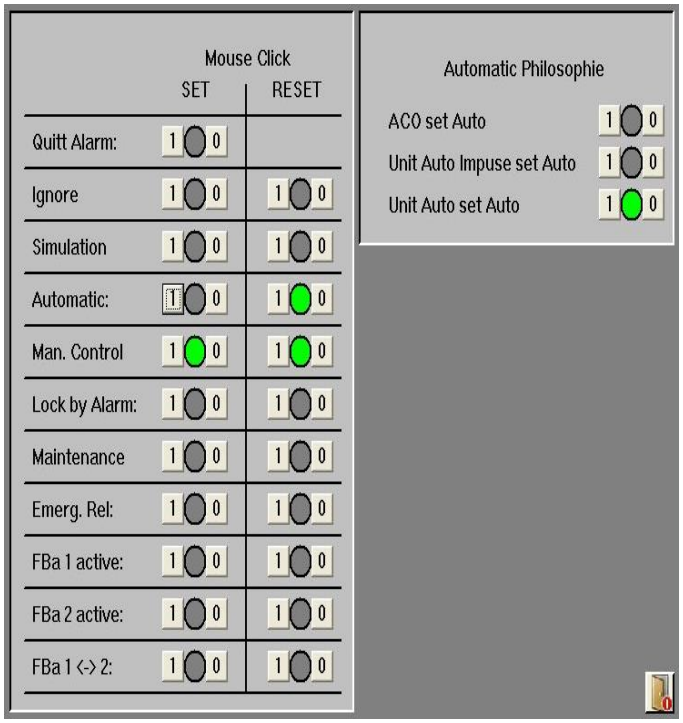
Assignment	Comment

FBaChange	Exchange inside and outside feedback 0 FBA1 = OFF = ON FBA2 1 = ON FBA2 FBA1 = OFF)
FBa1Active	Feedback 1 this present
FBa2Active	Feedback 2 is present
GAlQuitt	Recognition of an alarm (reset Gals)
IGN	Ignore alarm If an error message is sent o not
SIM	Simulation mode Feedback It is generated internally, the State is always OK
Auto	Automatic mode 1 = automatic 0 = manual
MCo	Manual control It is only effective in manual mode
EmRel	Emergency release Puenta a a production-related lock
InterlockGAl	Locked in alarm The actuator has alarm (GAL) which can not be operated.
Maint	Actuador en maintenance The actuator must not be actvado in manual or automatic There is no evaluation of failures
TOnSp	switching delay An automatic control is delayed by second xxx
TOfSp	the switch-off delay An automatic control extends to the second xxx
ADSp	Delay fault monitorings It starts each switching time. Yes the time expíRA, fault monitoring It has enabled.
TInterlock	Waiting time for reconnection The actuator is delayed until at least xxx seconds before a new Activation take effect.



7.5.Special configurations

In addition of the the window of the actuators, parameter settings are made by default, There is a window to the mouse setting. This determines what should happen when performs a click of the mouse on the element.
In addition in the parameterization of the mouse You can set ineven the behavior ManUAL/ Automatic generally:



- Control Automatic by actuator. If it powers an actuator, usually It is in automatic mode. Switching to manual mode is not always possible and when the actuator has effect program. This corresponds to the automatic philosophy of many programs (for example GEA) fermentation cellar.
- Side of unit AutoMatica (RUN) is only way ofl actuator in Automatic. Disabled the flank RUN in any momENTO can be activated manually.
- Auto drive (RUN) sets the mode Automatic of the actuator. While the corresponding unit is run cannot be changed in to the manual mode actuators.
- If one of these options This selected the operator the can stop manually at any time.
- The change of mode Manual a automatic mode It is always possible at any time.

7.6. Programming examples

Control for process Automatic

```
U "PH"                  While on this in this step and in "start"
"Act" s. Act [42]. ACo      42 actuator is activated in way automatic
"Act" s. Act [44]. ACo      44 actuator is activated in way automatic
```

Signal that enables automatic mode

```
U "RUN"                 process on "start"
"Act" s. Act [42] .xAuto    enables the mode actuator 42 automatic
"Act" s. Act [44] .xAuto    enables automatic actuator 44
```

External control

```
U "Din". DIN [15]. GIS      // Safety switch
"Act" s. Act [42]. ExCo     activated from an external control actuator
```

Release

Release of security

```
U "Din". DIN [11]. GIS      // Door Hmobre
"Din u". DIN [10]. GIS      Emergency stop
= "Act". Act [42]. Rel  // Conditional release of security
```

release process

```
U "Act". Act [40]. Off      // Valve off 1
U "Act". Act [41]. Off      // Valve off 2
= "Act". Act [42]. Rel2     // Release process-driven
```

Evaluation of alarm

```
U "Act". Act [42]. GAlS"          // Actuator in alarm
S    "HoldReq"                    // It keeps the unit
```

Assignment Act

From

```
U "Act". Act [1]. Out             signal to be activar output physics
=     A   0.0                     exit physics
A E 0.0                           //1 actuator feedback
= "Act". Act [1]. xFBa1           Active feedback 1
A E 200.0                         2 actuator feedback
= "Act"". Act [1]. xFBa2          Active feedback 2
```

Up to

```
U "Act". Act [1600]. Out          signal to be activate output physics
= TO 199.7                        exit physics
A E 199.7                         1600 actuator feedback 1
= "Act". Act [1]. xFBa1
AN E   399.7                      2 1600 actuator feedback
= "Act". Act [1]. xFBa2
```

7.7.Digital inputs (DIn)

All the DIn is considered as a simple digital inputs, It is not considered as a Feedback of actuators, door HomBres or empty signals.

The digital input associated with the input physics held in FC 502 "TransDIn". (E.g.employment in "Asignacion DIn").

7.8.Structure of data

Assignment	Type	Comment
EA0	BOOL	enable alarm by 0-signal
EA1	BOOL	enable alarm 1-signal by
SCS0	BOOL	status check alarm by 0-signal
SCS1	BOOL	check alarm by 1-signal status
xSig	BOOL	extern signal
B29	BOOL	spare
B30	BOOL	spare
B31	BOOL	spare
AlHM	BOOL	help memory for alarm
ImpHM	BOOL	help memory for impulse
xSigHM	BOOL	extern signal 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	alarm by 1 intern
ImpProt	BOOL	write impule flank to protocol
ImpNegProt	BOOL	write negative flank to protocol-impule
Switch	BOOL	convert as switch output
EA.	BOOL	General alarm
GAlS	BOOL	General alarm save
SCE	BOOL	status check error
GIS	BOOL	signal state
IMP	BOOL	boost flank
ImpNeg	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	turn off delay value
TOfDSp	REAL	turn off delay setpoint
ADVal	REAL	alarm delay current value
ADSp	REAL	alarm delay setpoint
SwCntVal	DINT	switch counter value

7.9.Commands

Assignment	Default	Comment
EA0	0	Alarmto in 0 - Sento the If the digital input is 0 will activate the alarm signal
EA1	0	Alarm in 1 - Sento the If the digital input is 1 is activated the alarm signal
SCS0	0	Checks the State of alarm in 0 - Sento the If the digital input is State 0It checks the status of the alarm
SCS1	0	Check the status of alarm 1 -Sento the If the digital input is State 1, checks the State of the alarm
xSig	x	Signal input physics This allocation is included in the standard program
User	x	Free for applications specific programmer

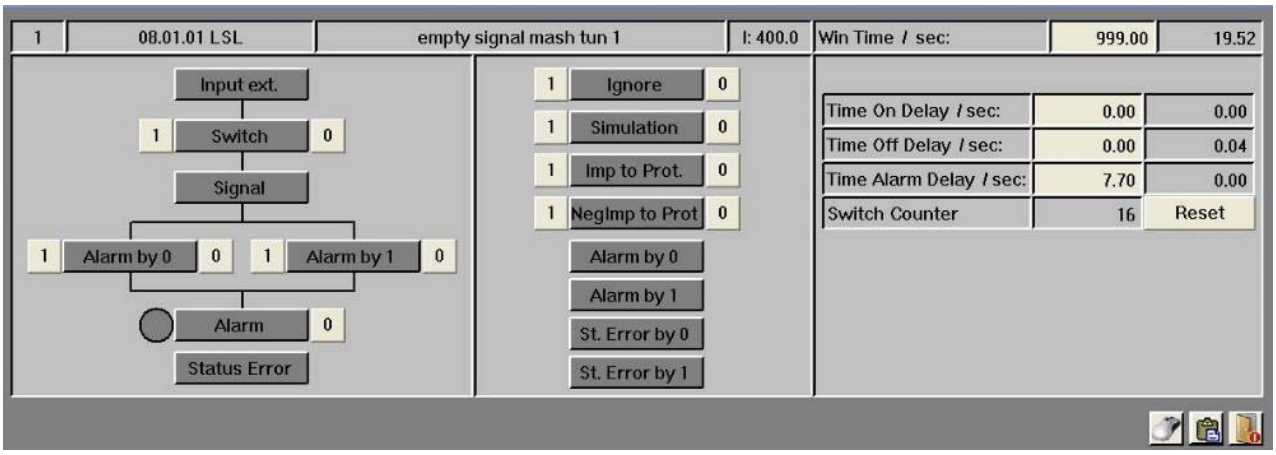
7.10. Status

Assignment	CpromoteIO
EA.	General alarm The State of the digital input is incorrect
GAlS	Stores General alarm General alarm was or is activated Signal must be reset by the operator (OK + Res)et alarm)
SCE	Designates an error condition State incorrect digital input for the next program start
GIS	Esignal state This signal contains programmable delays, so com the evaluation of a switch
IMP	Positive pulse of the digital input
ImpNeg	Negative pulse of the digital input

7.11. ParaMeasurements

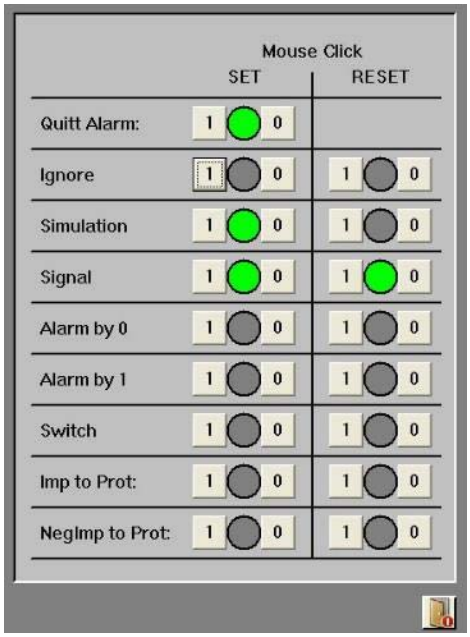
Bezeichnung	Kommentar
GAlQuitt	Recognition of an alarm (reset Gals)
IGN	Ignore alarm If an error message is sent o not
SIM	Simulation mode Feedback It is generated internally, the State is always OK
iEA0	Alarm signal 0 If the signal of the digital input ES 0, an alarm is triggered
iEA1	Alarm signal 1 If the signal of the digital input is 1 an alarm is triggered
ImpProt	Type o positive flankoperations mannualES With pulse positive for Register in the report of ooperations mannualES
ImpNegProt	Type o negative flankoperations mannualES With pulse negative for Register in the report of ooperations mannualES
Switch	Evaluation of the switch as digital input The status of the internal signal alternating with each positive flank .GIS
TOnDSp	Delayed change of signal from 0 to 1 If the entrance has the physical signal 1, is maniente internal signal by xxx seconds
TOfDSp	Delayed change of signal 1 to 0

	If the entrance has the physical signal 0, is maniente internal signal by xxx seconds
ADSp	Retardo of alarm If the signal is wrong, is delay xxx seconds in shootr the alarm



7.12. Configurations Special

In addition to the window of the sistema to the input digital para default settings of paraMeasurementsThere is the window to the mouse configuration. This determines what should happen when a click of the mouse on the element.



- You can simulate the digital signal of the sensor in case of problems (signal not critical to the process, such as a sensor of transport).
- Ignore alarms.

7.13. Examples of programming

CEW signal

```
U ' Din.Din [19]. " GIS"          // signal empty
S "PhaseEnd"                      // Finish step
```

Evaluation of alarm

```
A "Step0"                  // Stage is not 0
S "DIn". DIn [18]. EA0            // activate the alarm 0

U "DIn". DIn [18]. GAlS"          // Alarm
S "HoldReq"                // It keeps the unit
```

Cheartbeat of the State

```
"PA" U                            // Active step
S "DIn". DIn [12]. SCS0           // State activated with signal 0 error

A "DIn". DIn [12]. SCE"           // Without eerror state
S "PhaseEnd"                      // Finish step
```

AsigNAtion DIn

```
U E 400.0                  address input physics
= "DIn". [1] .xSig dIn     Active signal DIn 1 program
U E     599.7              address input physics
= "DIn". DIn [1600] .xSig         Active signal DIn 1600 program
```

8. Input analog (AIn)

When analog input is also any signal analog input with a result of the measuringsn, but estimatesndo values or values entered manually.
A calculated valuefor example, It is a capacity of a tank. Lpressure difference is first calculated on the basis of this differential pressure ofvolume of the tank based hl l. To view this content currently in the image of an analog input not It is possible, only displayed in final value.
The analog input associated with the input physics held in FC 503 "TransAIn". (Example in) "assignment AIn").
For the calculated values or numbers of input analostrategic should be used for a map with PEW for the reserves in the block of hardware.

8.1.Data structure

Asigncion	Tipo	CoMentarIO
SHE	BOOL	enable low low alarm
EHHA	BOOL	enable high high alarm
xAl	BOOL	alarm from extern
NPA	BOOL	not périphérie 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 limit - 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	HH enable alarm
iELLW	BOOL	enable LL warning
iEHHW	BOOL	enable HH warning
EA.	BOOL	General alarm
GAlS	BOOL	General alarm save
Warn	BOOL	General warning
Filter1	BOOL	filter 1 on (75%)
Filter2	BOOL	filter on 2 (88%)
Filter3	BOOL	filter 3 on (94%)
ManuInp	BOOL	manual input (not périphérie)
User	BOOL	memory free for user
PVal	REAL	process value
SP	REAL	setpoint
LScal	REAL	low scaling

HScal	REAL	high scaling
LLAVal	REAL	low low alarm value
LLVal	REAL	low low value (limit warning)
LVal	REAL	low value
Lavh	REAL	high value
HHVal	REAL	high value high (warning limit)
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 (without polygon) intern

8.2.Commands

Assignment	Default	Comment
SHE	0	Activation of Alarm low low To fall below the low low alarm limits, an alarm is triggered
EHHA	0	Activation of Alarm high high To the exceed the high high alarm limit is Active an alarm
xAl	0	Alarm externa If this signal 1 trips as an alarm
NPA	0	There are no peripherals In this caso, the value is taken to xPVal There is no scale conversion low o scale High
User	x	Free for specific applications programmer
xPVal	x	Process value Normally, this variable is supplied from the standard program If there are no periphery, so this value must be supplied by the programmer

8.3.Status

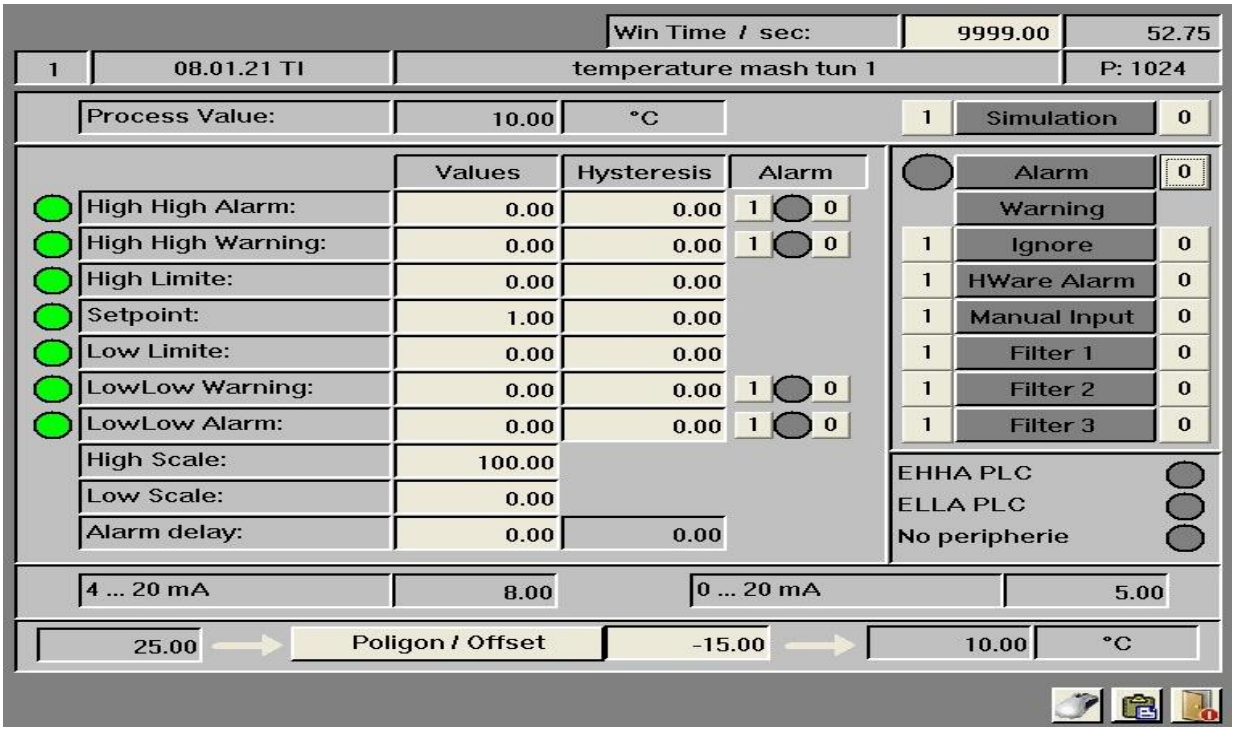
Assignment	ComEntarIO
MLLA	Low low alarm limits PVal > LLAVal => 1-SENAL A This value is assigned a histéreSIS
MLL	Limit Low low PVal > LLVal => 1-signal This value of allocates a hysteresis
ML	Limit under PVal > LVal => 1-signal This value is assigned a hiteresis
MSp	Desired value OK PVal > SP => 1 -SENAI

	This value is assigned a hiteresis
MH	Limit High PVal > Lavh => 1-Signal This value is assigned a hiteresis
MHH	Limit High high PVal > HHVal => 1-Signal This value is assigned a hiteresis
MHHA	Alarma Limit High high PVal > HHAVal => 1-Signal This value is assigned a hiteresis
MHWA	Alarm hardware (4-20 mA) Falls below the minimum limit (usually 2 mA) or exceed the maximum limit (usually 22 mA)
Gal	Alarma General Status of the analogue input is not well
GAlS	Stores General alarm General alarm was or is activated Signal must be reset por operator (OK + Reset) alarm)
Warn	Warning for operator If only for display purposes or for information of the entrthe system ADA.
PVal	Process value The converted value on physical drive

8.4.ParaMeasurements

Assignment	CpromoteIO
GAlQuitt	Recognition of an alarm (reset Gals)
IGN	Ignore alarm If an error message is sent o not
SIM	Simulation mode "PVal" value is manipulated directly on the display
iEHWA	Activates the alarm of monitoring hardware When the values of input physics is below or bails the limit
iELLA	Active Low low limit alarm is generated an alarm If "PVal" ES less than "LLAVal"
iEHHA	Active high limit alarm High is generated an alarm If "PVal" ES mayor What "HHAVal"
iELLW	Active low low limit warning is generated an alarm If "PVal" ES less than "LLAVal"
iEHHW	Active advertiencia of the high limit High is generated a warning if "PVal" ES mayor What "HHAVal"
Filter1	Filter 1 on (75%) The value of process filters. It makes sense in the case tickets oscillating
Filter2	Filter 2 in (88%) The value of process filters. It makes sense in the case tickets oscillating
Filter3	FILTr 3 en (94%) The value of process filters. It makes sense in the case tickets oscillating
ManuInp	Manual input (not peripheral) The analogue input is present virtually. Enters it the value of process the operator
SP	Set value
LScal	Scalel lie low Value physical the analog input minimum

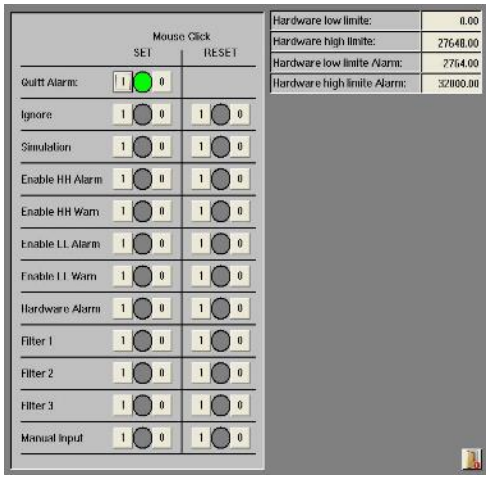
	Used to convert units of scale (mA) in the physical drive
HScal	Scalel lie high Value fisico the analog input maximum Used to convert units of scale (mA) in the physical drive
LLAVal	Alarmto of limit low low An alarm is generated if it falls of the limit under low
LLVal	Warning of low low limit An adevertenca is thrown if it falls of the limit under low
LVal	Limit low
Lavh	High limit
HHVal	Warning high limit high An adevertenca is thrown if it exceeds the limit high high
HHAVal	Alarm limit high high An alarm is generated if it exceeds the limit high high
LLAHys	Histeresis for alarm of Low low limit
LLHys	Histeresis for warning of Low low limit
LHys	Histeresis of limit Under
SpHys	Histeresis setpoint
HHys	Histeresis of limit high
HHHys	Histeresis for warning of limit High high
HHAHys	Histeresis for alarm of limit High high
ADSp	Alarm delay Yes the eState is incorrect an alarm is generated after xXX seconds
PoTNo	Positive value = number of polygonal table of conversion Negative value = process value twirl



8.5.Special configurations

In addition to the window of the sistema to the input analog para default settings of paraMeasurementsThere is the window to the mouse configuration. This determines what should happen when a click of the mouse on the element.

In addition to the parameterization of the mouseYou can determine the general scale of the entries:



- Hardware limit low - scale Division in 4 mA)(0 mA)
- Hardware limit High – Scale Division in 20 mA
- Hardware Alarm limit Under – Si the analog input cAE below this vAlor the alarm of yarn breakage.
- Hardware Alarma limit High – If the analog input exyields This value turns overflow alarm.

8.6.Programming examples

The value of process transfer

```
(L) "AIn". AIn [4]. PVal          // Temperature measurement
T "U002". [12]. Val               Unit parameter 12
```

Evaluation of alarm

```
A "Step0"                         // Stage is not 0
"AIn" s. AIn [18]. SHE                  // Activate the alarm of limit low low

U "AIn". AIn [18]. GAlS"                Alarm
S "HoldReq"                       // It keeps the unit
```

Assignment AIn

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

9. Regulador PID (PID)

Each analog output is supplied por a PID controller, but not always be has a PID for analog outputs. An example of a PID without analog output would be a cascade or a control with pulse output.
The PID associated to physical output is performed by the FC 504 “TransPID”. (Example in) "assignment PID”).
For control in Cascade or serial pulse outputsYou PID should be used which exists in the MaPA corresponding and lock the existing reserves in the hardware.

9.1.Data structure

Assignment	TIpo	CpromoteIO
GA.	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	active help memory starting value
Warn	BOOL	warning
GAlQuitt	BOOL	General alarm quitt
IGN	BOOL	ignore alarm
SIM	BOOL	simulation
Intermediate	BOOL	controller on mode (0 = off)
MSpExt	BOOL	setpoint extern mode (0 = intern)
DisOut	BOOL	disable output peripherie (0 = enable)
EW	BOOL	enable warning
B15	BOOL	spare
EA.	BOOL	General alarm
GAIS	BOOL	General alarm save
SCE	BOOL	status check error
Filter1	BOOL	filter 1 on (75%)
Filter2	BOOL	filter on 2 (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	REAL	setpoint extern
xPVal	REAL	process value from user programm
LScal	REAL	low scaling

HScal	REAL	high scaling
OVMin	REAL	output value min.
OVMax	REAL	output value max.
StC	REAL	static output value %
Strt	REAL	starting value %
StrTVal	REAL	starting time value
StrTSp	REAL	starting time setpoint
LLAVal	REAL	low value for alarm
HHAVal	REAL	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 boundary 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	comprehensive 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 gain 1
Fuzzy1V	REAL	Fuzzy variable 1
Fuzzy1VOld	REAL	Fuzzy old variable 1
Fuzzy2G	REAL	Fuzzy gain 1
Fuzzy2V	REAL	Fuzzy variable 1
Fuzzy2VOld	REAL	Fuzzy old variable 1
iOVal	REAL	output value intern
E	REAL	control error
DPart	REAL	derivative part

9.2.Commands

Asigacion	Default	CoMentarIO
GA.	0	Evaluation of activation of alarm
SCS	0	Activate status check
MStC	0	The regulator output setting to parameterize the value of output static (read-only)
MStrt	0	Home PID There are dyou ways of starting the controller 1. Setpoint controller startup time = 0 the output value initial is calculated once (Error * KP + starts the production of value) 2. Driver set home > 0 is written in the output of the PID, provided the home of parametrized initial value time has elapsed.
MOVMin	0	Adjustment of Output of the controller for parameterizing the minimum base
MOVMax	0	Maximum output value
OVOn	0	The regulator output setting to parameterize an maximum output value
User	x	Free for specific applications of the programmer
xSp	x	External set Point It is used by the program set pointIt is set by this value
xPVal	x	Value of external process It is used in process value
StC	x	Value of static output In some cases, this value has to be provided in the program Normally, the possibility of parameterization via the Visu is sufficient.
Fuzzy1V	x	Variable Fuzzy 1 It integrates a simple fuzzy logic, the variable fits shapesindo the maneuvers of control.
Fuzzy2V	x	Variable Fuzzy 2 It integrates a simple fuzzy logic, the variable fits shapesindo the maneuvers of control.

9.3.Status

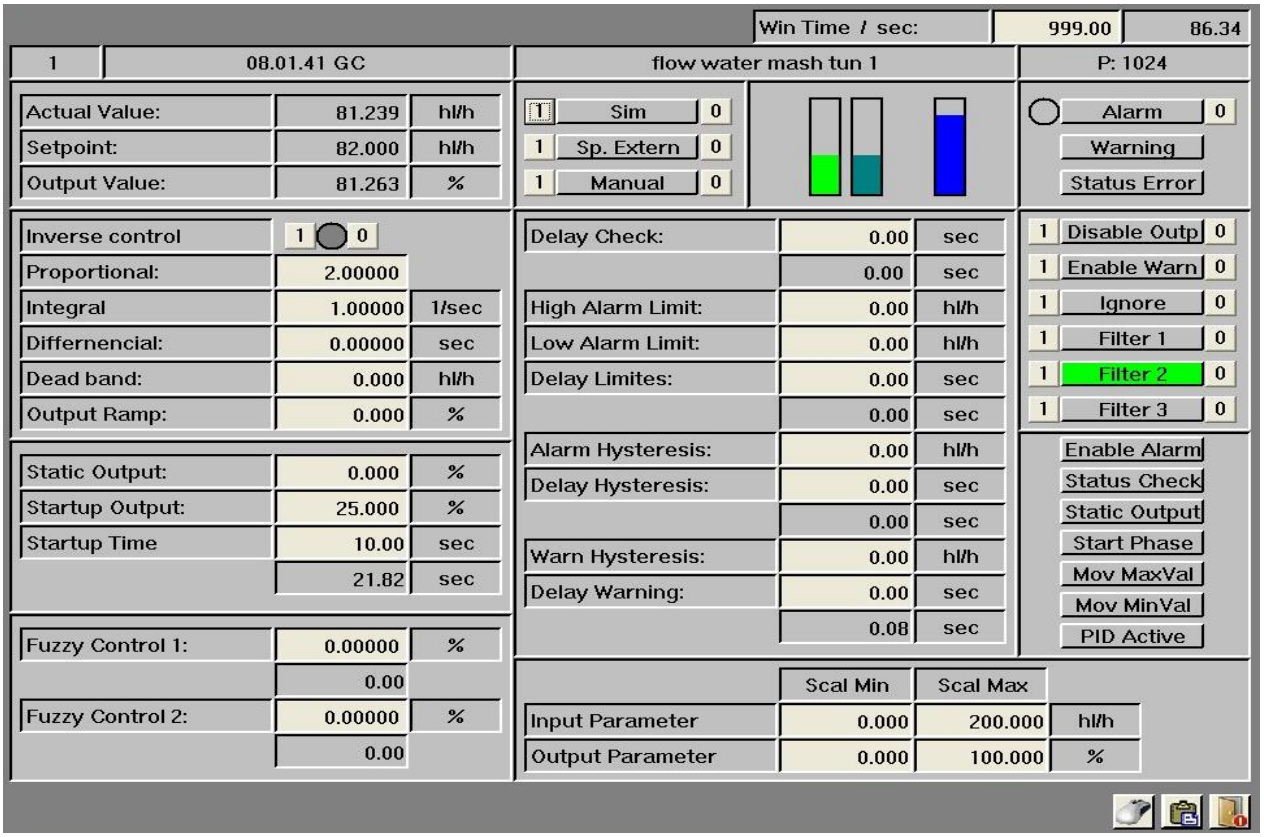
Assignment	CpromoteIO
Gal	Alarma General The State of the PID is not correct
GAlS	Stores General alarm General alarm was or is activated Signal must be reset por operator (OK + Reset) alarm)
SCE	Designates an error condition State wrong PID for the next program start

9.4.ParaMeasurements

Assignment	CoMentarIO
GAlQuitt	Recognition of an alarm (reset Gals)
IGN	Ignore alarm If an error message is sent o not
SIM	Simulation mode "PVal" value is manipulated directly on the display
Intermediate	Off control function (1 = off) If the control function is deactivated, the output value can write directly
MSpExt	Conigna point external (1 = external)

	The setpoint input value you can enter in the display
DisOut	Disable the output (1 = disabled) When the output is turned off, the PID output value is not transmitting to the periphery
EW	Activate alarm warning If the value of proaccess is off limits is generated with warning
Filter1	Filter 1 on (75%) The value of process filters. It makes sense in the case tickets oscillating
Filter2	Filter 2 in (88%) The value of process filters. It makes sense in the case tickets oscillating
Filter3	FILTr 3 en (94%) The value of process filters. It makes sense in the case tickets oscillating
CA	Control direction (1 = reverse) This bit of the control direction can be reversed.
Ovto the	Output PID value If you select remote control function, the output value can be inserted directly.
SP	Consigna PID Sand it can be controlled externally the It recorded, the slogan can be inserted directly
PVal	Process value In simulation mode, the value of process can be entered directly (for testing only)
LScal	Scale Low Physical minimum value of the analog input Only for the visualization of trend, has no effect on the function of the driver
HScal	Scale High Physical maximum value of the analog input Only for the visualization of trend, has no effect on the function of the driver
OVMin	M-valueminimal output of the PID With the function activated the output of the PID controller is limited to this value
OVMax	Maximum value Output of the PID With the function activated the output of the PID controller is limited to this value
StC	Value of output static (read-only) Static output driver of this value mode writes to the output of the driver
Strt	Initial value for the control While the start-up phase of operation is active, This value is written in the controller output In a reference time of 0 seconds for the start-up phase, This value is used as a compensation to the initial calculation of the output of the controller.
StrTSp	Time adjustment ofcontroller I in start-up With set points > 0 sec the initial value of the driver is written in the output of the driver
LLAVal	Alarma minimum limit If it falls below the value of process, this willLor, an alarm is triggered (t(eniendo into account the delay times)
HHAVal	Alarm threshold If this value is exceeded by the value of the process, an alarm is triggered (Taking into account the delay times)
AHys	Alarm hysteresis The process is outside this hysteresis (difference between) Setpoint and actual value) An alarm is triggered (taking into account the delay times)
CheckDSp	Delay activation of alarms While this timer is still running, the evaluation of alarm is not active

ADHLSp	Delay alarm when limits are exceeded This time, an alarm can be delayed when limits are exceeded
ADSp	Hysteresis alarm delay exceeds This time, an alarm may be delayed due to overcome the hysteresis.
WHys	Hysteresis warning Aalert for hysteresis
WDSp	Superor delay hysteresis warning At that time, a warning may be delayed due to overcome the hysteresis.
KP	The control provides
KI	Control Integral
KD	Differential Control
RampV	entrance ramp Maximum change in output per second to the point of setting = 0, this value is inactive
DeadB	Deadband control deviation When the process value is within the dead band, the output of the regulator freezes
Fuzzy1G	Fuzzy gain 1 Weighting the first fuzzy logic
Fuzzy2G	Gain 2 Fuzzy Weighting the first fuzzy logic



9.5.Special configurations

In addition to the window of the sistema to PID PAra default settings of paraMeasurementsThere is the window to the mouse configuration. This determines what should happen when a click of the mouse on the element.
In addition to the parameterization of the mouseYou can determine the general scale of entries:

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

9.6.Programming examples

Transfer of securities

```
(L) "Uxx". [12]. Val          // parameter 12 unit - temperature measurement
L    "PID". PID [4] .xPVal          Process value for PID

(L) "Uxx". [12]. SP           // parameter unit 12- Temperature measurement
(L) "PID". PID [4] .xSp                  Nominal value of PID
```

Evaluation of alarm

```
U    "Act.Act [45]." Out"          Output of the be sure to actor
A "CIP"                            CIP
S "PID". PID [4]. GA.              //Enable alarm limit low

"PID" U. PID [4]. GAlS"            Alarma
S "HoldReq"                   // It keeps the unit
```

Home / Static output of PID

```
U ' Act.Act [45]. " Out"           // output of the be sure to actor
U "CIP"                            CIP
S "PID". PID [4]. MStC             // Static output active

U ' Act.Act [45]. " Out"           // Output of the actuator
S    "PID". PID [4]. MStrt         It starts the PID
```

PID assignment

```
L 'PID". PID [1]. Ovto the         output value of PID 1
L #OutFactor                       they are 327.67
* R                                multiplier
RND                                rounds
T PAW 1024                    you pass the value to output physics
(L) "PID". PID [480]. Oval         output value of the PID 480
L #OutFactor                       they are 327.67
* R                                multiplier
RND                                rounds
T PAW 1982                    you pass the value to output physics
```

10. Modulor counter (Cnt)

Module timer (count) is used to add certain values of pulyou're in the cyclic sequence. For the PLC there is available up to 320 Measurements, It is more than enough. An allocationn set of peripherals and number modulo counter does not exist here.
A often ES useful to divide a flow meter in hardware in several modules of timer in the software. An example would be a water flowmeter. In order to calculate the amount of water What was pumped into the container 1as well as integrating an online consultation, the solution cleaner is cofigurl'm a counter for each line.
Also the value that means each pulse delivered to the counter must be adjusted.

10.1. Esdata infrastructure

Assignment	Type	CpromoteIO
EAImp	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 limit - warning if enabled
MHHA	BOOL	high high alarm - alarm if enabled
ImpHM	BOOL	impulse memory help
GAlQuitt	BOOL	General alarm quitt
IGN	BOOL	ignore alarm
SIM	BOOL	simulation
Reset	BOOL	Reset counter
iELLA	BOOL	counting book
iEHHA	BOOL	HH enable 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	boost flank
B04	BOOL	spare
B05	BOOL	spare
B06	BOOL	spare
User	BOOL	memory free for user
PVal	REAL	process value
SP	REAL	setpoint
LScal	REAL	low scaling
HScal	REAL	high scaling

LLAVal	REAL	low low alarm value
LLVal	REAL	low low value (limit warning)
LVal	REAL	low value
Lavh	REAL	high value
HHVal	REAL	high value high (warning limit)
HHAVal	REAL	high high alarm value
ADVal	REAL	alarm delay value
ADSp	REAL	alarm delay setpoint
ImpVal	REAL	value per impulse
CVal	DINT	counter value

10.2. Commands

Assignment	Default	Comment
EAImp	0	Monitoring the pulse of activation Monitored if at any time There has been a striking If not, an alarm is triggered
SHE	0	Activation of Alarm low low To fall below the low low alarm limits, an alarm is triggered
EHHA	0	Activation of Alarm high high To the exceed the high high alarm limit is Active an alarm
xAl	0	Alarm externa If this signal 1 trips as an alarm
ResetBlock	0	Lock reseteo If this signal This 1, the counter may not be reset
xSig	0	External pulse signal At this sign of digital input of the hardware are you assign to a counter
User	x	Free for specific applications of the programmer

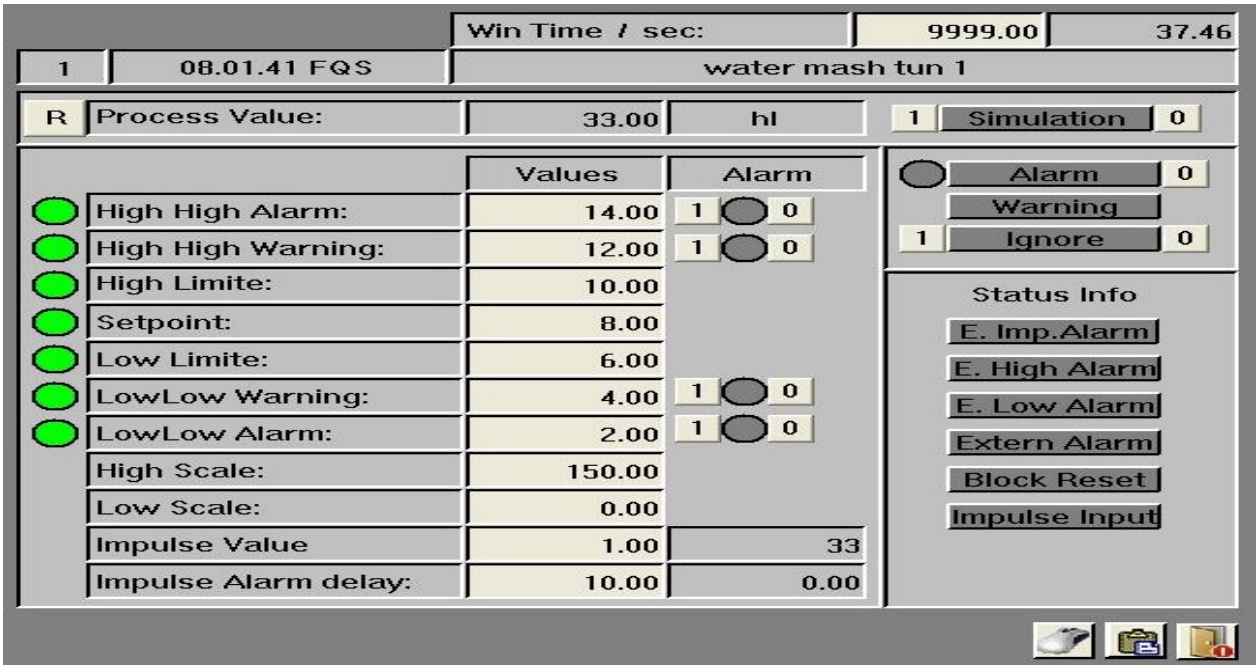
10.3. Status

Assignment	Comment
MLLA	Low low alarm limits PVal > LLAVal => 1-SENAL This value is assigned a hysteresis
MLL	Limit Low low PVal > LLVal => 1-signal This value of allocates a hysteresis
ML	Limit under PVal > LVal => 1-signal This value is assigned a hiteresis
MSp	Desired value OK PVal > SP => 1 -SENAl This value is assigned a hiteresis
MH	Limit High PVal > Lavh => 1-Signal This value is assigned a hiteresis
MHH	Limit High high PVal > HHVal => 1-Signal This value is assigned a hiteresis
MHHA	Alarma Limit High high PVal > HHAVal => 1-Signal

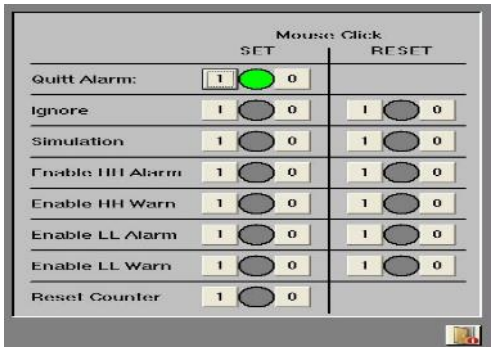
	This value is assigned a hiteresis
EA.	Alarma General Status of the analogue input is not well
GAIS	Stores General alarm General alarm was or is activated Signal must be reset por operator (OK + Reset) alarm)
Warn	Warning for operator If only for display purposes or for information of the entrthe system ADA.
IMP	flank A positive flank it is formed from the signal xGIS.
PVal	Process value

10.4. ParaMeasurements

Assignment	CoMentarIO
GAlQuitt	Recognition of an alarm (reset Gals)
IGN	Ignore alarm If an error message is sent o not
SIM	Simulation mode "PVal" value is manipulated directly on the display
Reset	Reset counter The counter is reset to zero
iELLA	Active low low limit alarm is generated an alarm If "PVal" ES less than "LLAVal"
iEHHA	Active high limit alarm High is generated an alarm If "PVal" ES mayor What "HHAVal"
iELLW	Active low low limit warning is generated an alarm If "PVal" ES less than "LLAVal"
iEHHW	Active advertiencia of the high limit High is generated a warning if "PVal" ES mayor What "HHAVal"
SP	Nominal value
LScal	Scalel lie low Value physical the analog input minimum Used to convert units of scale (mA) in the physical drive
HScal	Scalel lie high Physical valueo the analog input maximum Used to convert units of scale (mA) in the physical drive
LLAVal	Alarmto of limit low low An alarm is generated if it falls of the limit under low
LLVal	Warning of low low limit An adevertenca is thrown if it falls of the limit under low
LVal	Limit low
Lavh	High limit
HHVal	Warning high limit high An adevertenca is thrown if it exceeds the limit high high
HHAVal	Alarm limit high high An alarm is generated if it exceeds the limit high high
ADSp	Alarm delay Yes the eState is incorrect an alarm is generated after xXX seconds.
ImpVal	Value of a pulse Do not enter a pulse of the hardware Example, weight of malta: 1 impulse = 50 kg; Thus ImpVal = 50



10.5. Special configurations



In addition to the window of the sistema to counter para default settings of paraMeasurementsThere is the window to the mouse configuration. This determines what should happen when a click of the mouse on the element.

10.6. Programming examples

Signal transfer

```
U "Act". Act [18]. Out          // Vservice water
U "DIn". DIn [233]. IMP         // positive input ditital flank
= "Cnt". CNT [4] .xSig          // external signal for counter
```

Transfer of securities

```
(L) "Cnt". CNT [4]. PVal        water meter
T    "Uxx". [17]. Val       Unit parametro 17
```

Evaluation of alarm

```
U "Act". Act [18]. Out          Water seal
S "Cnt". CNT [4]. EAImp         // It enables alarm pulse

U "Cnt". CNT [4]. GAlS"         Alarma
S "HoldReq"               // It keeps the unit
```

11. Modulor messageboard (Msg)

Given that the other modules as actuators, digital inputs, etc already. integratesn messagesvery rarely used This module of mensaje. SOLO is used to generate alarms or instructions for the operator that are not directly related to the periphery.

11.1. Esdata infrastructure

Assignment	TIpo	CoMentarIO
B24	BOOL	Sstop
B25	BOOL	Sstop
B26	BOOL	Sstop
B27	BOOL	Sstop
xAlarm	BOOL	extern signal for alarm condition
B29	BOOL	Sstop
B30	BOOL	Sstop
B31	BOOL	Sstop
B16	BOOL	Sstop
B17	BOOL	Sstop
B18	BOOL	Sstop
B19	BOOL	Sstop
B20	BOOL	Sstop
B21	BOOL	Sstop
B22	BOOL	Sstop
B23	BOOL	Sstop
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
EA.	BOOL	General alarm
GAIS	BOOL	General alarm save
OPMsgActive	BOOL	operator message active
AlarmMsgActive	BOOL	active alarm message
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

11.2. Commands

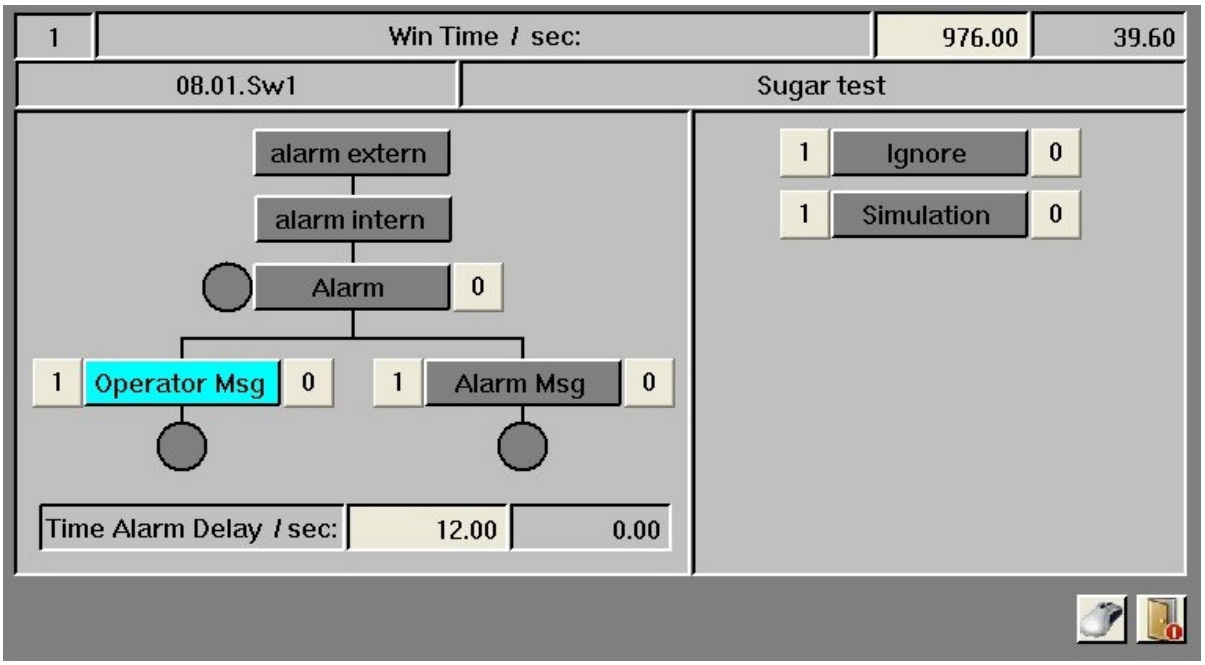
Assignment	Default	CoMentarIO
xAlarm	0	The activation of the message)Message) If this signal is 1,is generated a message (or alarm) operator
User	x	Free for specific applications programmer

11.3. Status

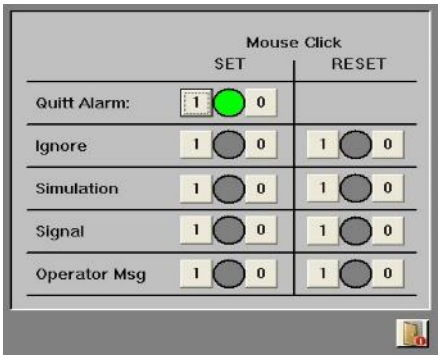
Assignment	CoMentarIO
EA.	Alarma General Status of the analogue input is not well
GAIS	Stores General alarm General alarm was or is activated Signal must be reset por operator (OK + Reset) alarm)
OPMsgActive	Active The operator message Note operator is active.
AlarmMsgActive	Active alarm message The alarm is active.

11.4. ParaMeasurements

Assignment	ComEntarIO
GAIQuitt	Recognition of an alarm (reset Gals)
IGN	Ignore alarm If an error message is sent o not
SIM	Simulation mode "PVal" value is manipulated directly on the display
OPMsg	Message of operator OPMsg = 1 an operator message is generated when the signal is active xAlarm OPMsg = 0, an alarm message is generated when the signal is active xAlarm
ADSp	Alarm delay (message) Yes the eState is incorrect an alarm is generated after xXX seconds.



11.5. Special configurations



In addition to the window of the sistema to counter para default settings of paraMeasurementsThere is the window to the mouse configuration. This determines what should happen when a click of the mouse on the element.

11.6. Programming examples

Generate message

```
U "Malzlaster Glaube"          // Malta this truck waiting
= "Msg". MSG [12] .xAlarm      // External signal for the message
```

Evaluation of alarm

```
U "Msg". MSG [12]. GAlS        // Active message
S "SignalLamp"                 // visual signal for the operator
```

12. Isoftware switch (Switch)

In order to achieve an s operationsimple and uniform, the module Switch (switchsoftware Tor) is integrated to the operator to generate a signal to PLC (independent of other modules), the Switch by default most used would be the button to "confirm alarm" or confirm an operation manual.

12.1. Esdata infrastructure

Assignment	Tipo	CpromoteIO
Set	BOOL	set software switch
Reset	BOOL	reset switch software
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

12.2. Commands

Like thisgnation	Default	CoMentarIO
Set	0	Serves as a switch in the program sequence While this signal is 1, it is also featured in the display, to the switches It can be operated.
Reset	0	It serves as a switch to go back in the program sequence While this signal is 1, it is also featured in the display, that switch can not be operated.
User	x	Free for specific applications of the programmer

12.3. Status

Assignment	CpromoteIO
GIS	Condition State of the switch software

12.4. Programming examples

Restart of Switch

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

Consultation of the status of the Switch

```
U "PH"
U 'Switch'. Switch [3]. GIS          confirmation of the operator 'manual emptying of sugar'
"Act" s. Act [42]. TOco              fodder mixer
```

13. UniDad (unit)

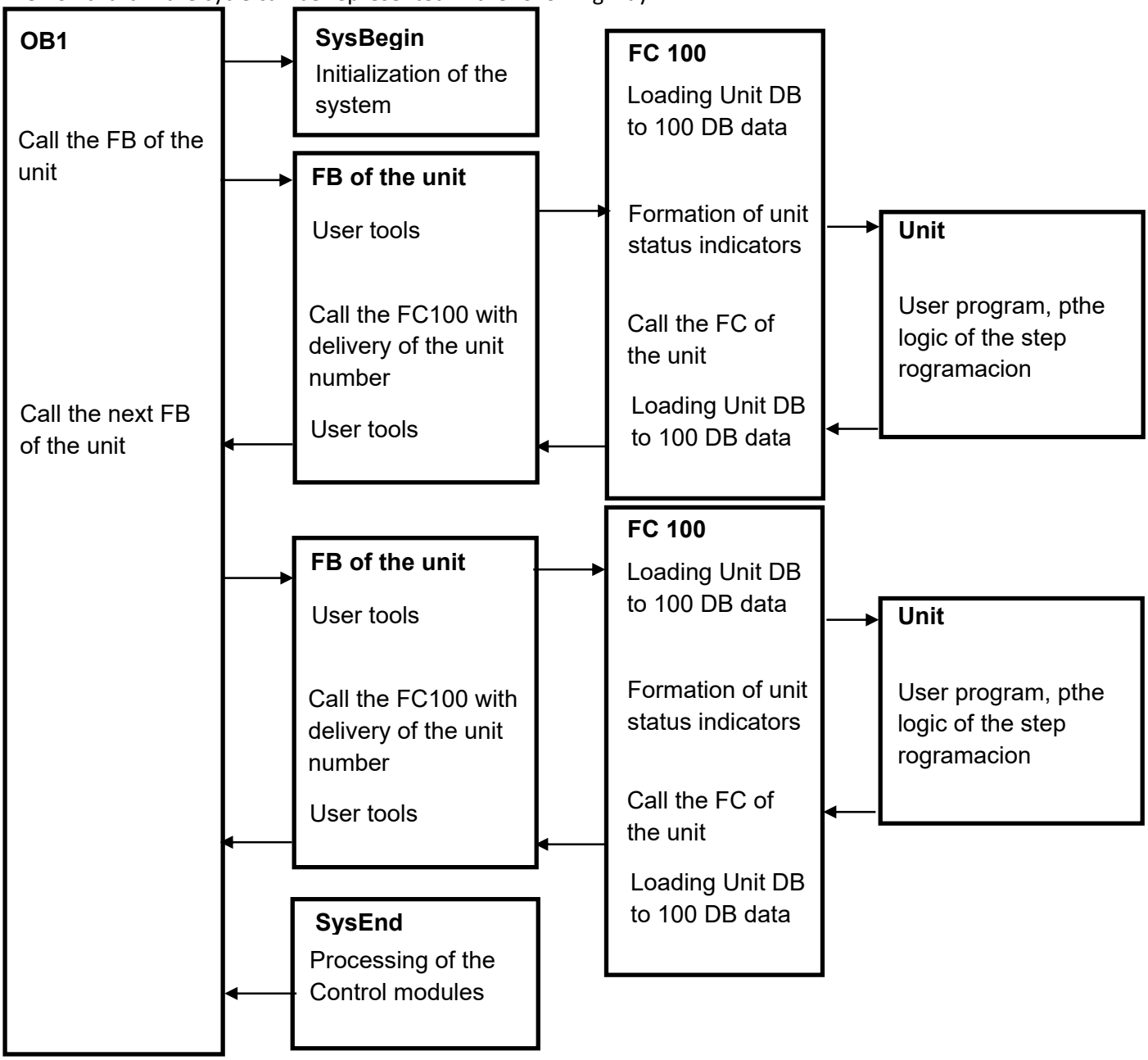
"Unit" is called a unit of production, such as tank fermentation, pasteurizers, or filter, called in earlier systems and sequencer. The programming of the units is a real and central task. In order to achieve aa quick and easy programming provided by a large number of tools.

Basically, a unit is always a DB, a FB and a FC that is assigned on a permanent basis. Expiration dates and recipe values are stored in DB. FB fundamentals assignments are completed and the function of the Unit-master (FC 100) block is called. En FC, the pa can be programmedsos (activities, actions).

The distribution is as follows:

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

The flowchart in the cycle can be represented in the following way.



Basically this diagram represents the implementedn that was made in a cycle:

- first begins with the OB1 donde is called the unit BF
- In the FB locks actuators, transferncia of process values and paraMeasurements are performed, he is called the FC of the same unit.
- The FC are the execution to be carried out at each step, be reset counters, etc.

13.1. End of a step

Is terminated when a step of a recipe in a unit (PhaseEnd = TRUE)the following actions are performed in the same cycle without interruption.

- The current step executes and detects the final status = 1
- The step current is processed again with identifier "PLCycle" (last cycle)
- The income the current data in the registry
- Is loaded lvalues of the recipe
- Se update status indicators and paraMeasurements modules
- Is processing new step with identifier "PFCycle" (first cycle)

14. Communication unit to unit

One of the more elaborate ES incronizar the units together. To perform this task rapidamenThere you are a standardized interface provided by BatchXpert. Each unit has the possibility to communicate simultaneouslysimultaneously with 4 teachers and 4 slaves. It is intended for in 4 maestrI or 4 slaves of different kinds.

Example:

The exchange of data ofl cooling grape juice to 99 fermentation tanks occupied a single level. Therefore the cooling of the grape juice has as “Slave 1” a fermentation tank. The composed of 2 slaves can be a tank of yeast or propagator etc.

14.1. Specification of the Name

- UnitCom.U.Master1.xxx //Signals the current unit in Master 1
- UnitCom.Master1.xxx //Signals ofl Master 1 in the current drive
- UnitCom.U.Slave1.xxx //Signals the current drive in the Slave 1
- UnitCom.Slave1.xxx // Signals ofl Slave 1 in the current drive

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

14.2. Commands

Assignment	Default	Comment
UnitCom.U.Master1.TransReq	0	Application for transfer
UnitCom.U.Master1.TransActive	0	Active transfer
UnitCom.U.Master1.TransEnd	0	Transfer finished
UnitCom.U.Master1.TransRel	0	Free transfer control
UnitCom.U.Master1.FillReq	0	Lrequested lenado
UnitCom.U.Master1.FillActive	0	LActive lenando
UnitCom.U.Master1.FillEnd	0	Finished filling
UnitCom.U.Master1.FillRel	0	Libera control filling
UnitCom.U.Master1.FunctionNo	0	number of special function For withoutcronizaciones of complex flow
UnitCom.U.Slave1.TransReq	0	Application for transfer
UnitCom.U.Slave1.TransActive	0	Active transfer
UnitCom.U.Slave1.TransEnd	0	Transfer finished
UnitCom.U.Slave1.TransRel	0	Free transfer control
UnitCom.U.Slave1.FillReq	0	Lrequested lenado
UnitCom.U.Slave1.FillActive	0	LActive lenando
UnitCom.U.Slave1.FillEnd	0	Finished filling
UnitCom.U.Slave1.FillRel	0	Libera control filling
UnitCom.U.Slave1.Connect	0	Connect request
UnitCom.U.Slave1.Start	0	Start unit
UnitCom.U.Slave1.PrIdChange	0	Change of PrId
UnitCom.U.Slave1.FunctionNo	0	number of special function For withoutcronizaciones of complex flow
UnitCom.U.Slave1.No	x	Number of slave unit PARA maneuver unit number of Associate
UnitCom.U.Val1	x	nominal value or the vthe associated drive real value
UnitCom.U.Val2	x	nominal value or the real value of the associated unit
UnitCom.U.Val3	x	nominal value or the real value of the associated unit

UnitCom.U.Val4	x	nominal value or the real value of the associated unit
UnitCom.U.Val5	x	nominal value or the real value of the associated unit
UnitCom.U.Val6	x	nominal value or the real value of the associated unit
UnitCom.U.Val7	x	nominal value or the real value of the associated unit
UnitCom.U.Val8	x	nominal value or the real value of the associated unit

14.3. Status

Assignment	CoMentarIO
UnitCom.Master1.Used	Used unit
UnitCom.Master1.OpRequest	The operator request
UnitCom.Master1.Steril	Sterile unit
UnitCom.Master1.Clean	clean unit
UnitCom.Master1.NotClean	Unit Dirty
UnitCom.Master1.Product1	product 1
UnitCom.Master1.Product2	product 2
UnitCom.Master1.Product3	Product 3
UnitCom.Master1.Product4	product 4
UnitCom.Master1.ReqCIP	last stage
UnitCom.Master1.GAlQuitt	Reset general alarm
UnitCom.Master1.Ign	ignore the alarm
UnitCom.Master1.Sim	Simulacion
UnitCom.Master1.Run	Unit in run mode
UnitCom.Master1.Pause	Unit in pause mode
UnitCom.Master1.Hold	Unit in standby mode
UnitCom.Master1.EmHold	Spare - it keeps emergency in the unit
UnitCom.Master1.Maint	Maintenance
UnitCom.Master1.GAl	Alarma General
UnitCom.Master1.GAlS	Stores AlarMA General
UnitCom.Master1.SCE	It says an error condition
UnitCom.Master1.Watchdog	alarm monitoring
UnitCom.Master1.Step0	Unit in step 0
UnitCom.Master1.ReadyStart	Ready to start unit
UnitCom.Master1.Active	Active unit (not in step 0)
UnitCom.Master1.CIPModus	Unit in CIP mode
UnitCom.Master1.TransReq	transfer request
UnitCom.Master1.TransActive	transfer active
UnitCom.Master1.TransEnd	The end transfer
UnitCom.Master1.TransRel	release control transfer
UnitCom.Master1.FillReq	requested filling
UnitCom.Master1.FillActive	filling active
UnitCom.Master1.FillEnd	End of filling
UnitCom.Master1.FillRel	Release control filling
UnitCom.Master1.Connect	connected
UnitCom.Master1.Start	Home of Unit Associated

UnitCom.Master1.PrIdChange	PrId the associated change The associated unit starts with a new PrId For example, in a change of range However, it is always advisable to use a system specially adapted to a change of range
UnitCom.Master1.FunctionNo	Number of special function For the synchronization of complex flow
UnitCom.Master1.UnitNo	Associated unit number
UnitCom.Master1.PrId	PrId
UnitCom.Master1.Charge	Lot number Production number
UnitCom.Master1.ProgNo	program number Normally written in identification of the product
UnitCom.Master1.Val1	value 1
UnitCom.Master1.Val2	Value 2
UnitCom.Master1.Val3	value 3
UnitCom.Master1.Val4	value 4
UnitCom.Master1.Val5	value 5
UnitCom.Master1.Val6	value 6
UnitCom.Master1.Val7	Value 7
UnitCom.Master1.Val8	value 8

Signals and additional values to masters and slaves must match to schematic

14.4. Programming examples

Conectar with slave

Lcommunication can be set to a slave, If you pass a the number of the escl unitAVO, y set the "Connect" signal. For most of the compounds, the unit number slave You can transfer statically, since this does not change. Example of a line of production straight or with a line of development of the brewery. The number of slave unit may also be dynamically. Dand according to certain criteria of the program, is can usefulIZAR altogether the grape juice 1 boiler or 2 dICHA dynamic selection, however, can be taken by a code system (Production, calendar of system production planning system).

```
A "Step0"
S "UnitCom". U.Salve1.Connect
L 13                                    number of the associated unit = 13
T "UnitCom". U.Salve1.No
```

One-step synchronization flow

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

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

Release for triggers

By controlling thes active components for the release of transferencia must always be consulted (Filling (or release). Therefore, a mutual closing in case of breakdown or if an operator sets the retention of Unit with ease.

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


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

15. Module parameter of Unit

For the transfer of the values do not: recipe of the BatchXpert, or the registration of the flow values, is usedZan paraMeasurements. There are paraMeasurements of 1 to 40. A maximum of 16 parametrYou can be transferred by step. It should be noted that the parameter 1 always ES used as a temposurveillance of the step curler. For el supply modules with values nominal and parameterization automatically from the standard program.

15.1. Structure data

Assignment	Type	CpromoteIO
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
Not	BYTE	ParamodulNo
OnlySp	BOOL	only setpoint
OnlyVal	BOOL	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	manual input requiered
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

15.2. Commands

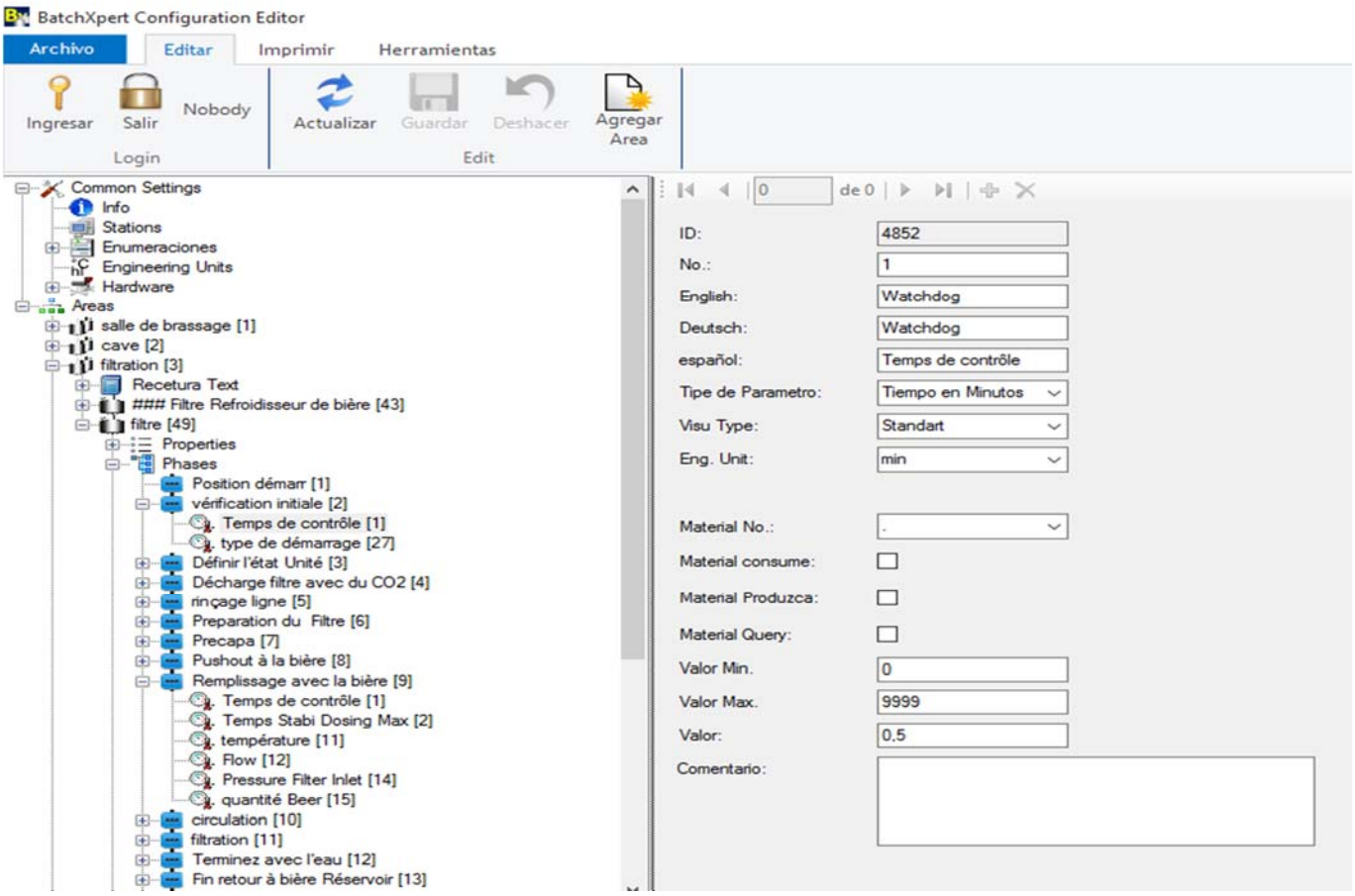
Assignment	Default	ComEntarIO
S	0	It will be used only if the parameter module is configured as a time signal, 1 = Weather started 0 = the signal value is reset to zero
H	0	It will be used only if the parameter module is configured as a time signal, 1 = time stops
Reset	0	Restart of module paraMeasurements
Val	x	Actual parameter module This variable is the real value of maneuver When configured as a moment that has been not classified

15.3. Status

Assignment	CpromoteIO
OK	Value OK Actual value (Val) > = nominal value (Sp)

15.4. To configure the configuration editor and recipe

Like thisgnation	CoMentarIO
Not	Number of module parameter Awarded the corresponding number, which is also used in the program
OnlySp	Sand used only as a point of nominal value of the module's paraMeasurements
OnlyVal	Sand it is only used as an actual value modulo's parameter
Enum	Enumeration For setpoint texts of the screen are used Example: agitator OFF u ON
TSec	Configuration of the module's time at second base
TMin	Configuration of the module's time on the basis of Minutes
THour	Time module configuration on the basis of hours
TDay	Time module configuration on the basis of day
Endcond	Final condition of the module paraMeasurements Single color on the screen of display
ManuInput	Manual input to the module using It is a codigo's color on the display The operator can manually enter the value
AlarmCond	A module of parametrs alarm condition (Alarm limit) Single color display screen
SP	Setpoint for module paraMeasurements



Example of a configuration of the module parameter in the "BatchConfigurator"

15.5. Change set at run time

The setpoint of the modules paraMeasurements could void at runtime using the window ls Unit. Of course, this type of action can be detected, for the Protocol to manual intervention.

15.6. Programming examples

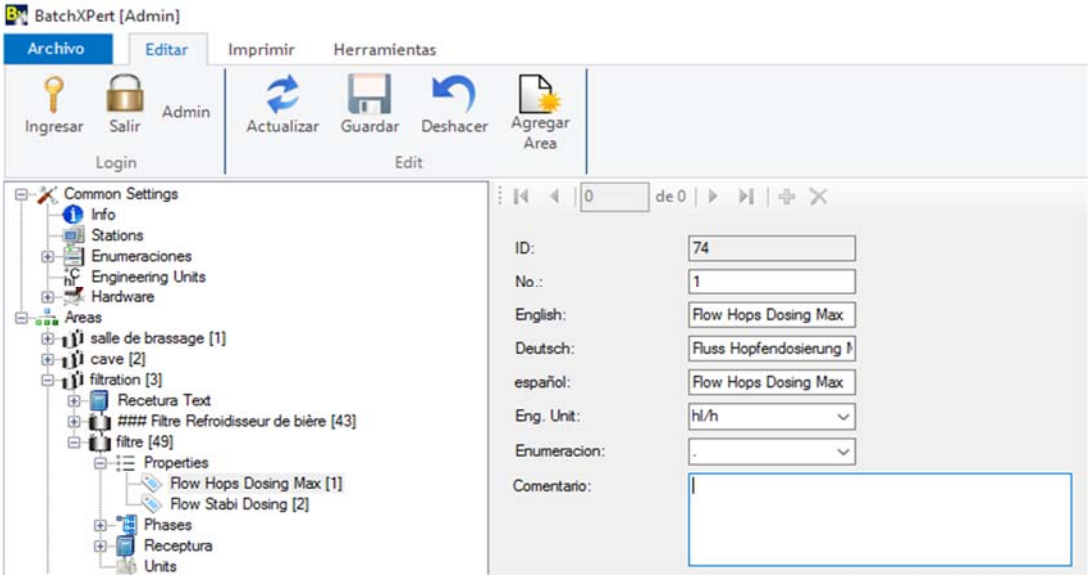
```
"PA" U
S "Uxx". [2]. S          starts the module paraMeasurements

A "DIn". DIn [12]. GIS          LSL
ON "Act". Act [32]. Out         // Pump
ON "PH"
S "Uxx". [2]. H

U    "Uxx". [2]. D      First or exceeded the setpoint from the value of the parameter.
 = "PhaseEnd"                End of step
```

16. Property of the UniDad

Settings or properties of a special unit can be stored in the properties of the unit.

16.1. Configuration (BatchConfigurator)

Configuration)TeX(the ad to, unit,...) This fact at the level of classs in the configuration editor



16.2. Enter the values for each unit

The supply of the variable with values is carried out through of the display unit en 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

16.3. Programming examples

A property of the unit is associated with a unit to calculate the amount of localized.

```
(L) "Uxx". Property [2]                    // volume tubing tank
T "UnitCom". U.Val1
```

17. Status indicators of the Unit

indicators ofl Unit status they serve as additional programming help. These variables they can be read if a unit is Hold, if I turn it It has run through the first cycle. Also reports on these variables if ato failure is present, the proceso must be stopped or The endperform the step.

17.1. Commands

Symbol	Operando	CoMentarIO
HoldReq	4.0 M	Manter request Designates a fault is present, stop the production process
ShowAlarm	4.1 M	Unit in alarm State Designates a fault is present, stop the production process
OpReq	4.2 M	operator request Expected action by the operator, It signalsdo in the display
ProtWrite	4.3 M	writing ofl Protocol step The step data ACTUthe (State, Setpoint and values reals) Sand they are stored in the registry writing. It is used only for special registration tasks. Change of step or status change are automatically written data in the registry.
CIP	4.4 M	Unit in CIP mode Indicates that a toilet ESTa Active
PhaseEnd	4.7 M	end of step Indicates who is alcAnza final step condition In "RUN" mode the program jumps to the next stage
StatusInfo	MB 1	Unit status information This variable shows the current drive What You can describe the status See blemishes 1, 0-1 7 (sterile, clean, used, ...)
StepNoNew	MB 9	Nnew number of step (hop) They have to connect the end of the stage in the next step, but skips to a specific addressFICA in the recipe, so be transfer the amount desired values ofl step.

17.2. Status

Symbol	Operando	CpromoteIO
StatusSteril	1.0 M	Sterile unit status Unit is sterile
StatusClean	1.1 M	Clean unit status Unit is clean it
SatusNotClean	1.2 M	Unit status is not clean The unit is not clean
StatusProd1	1.3 M	Unit status of the Product 1 Unit was used with the type 1 product
StatusProd2	1.4 M	Unit status of the Product 2 Unit was used with the type 2 product
StatusProd3	1.5 M	Unit status of the Product 3 Unit was used with the type 3 product
StatusProd4	1.6 M	Unit status of the Product 4 Unit was used with the type 4 product
ReqCIP	1.7 M	Unit status applies for CIP

		It must be cleaned prior to the start of the next unit
Run	2.3 M	Unit in "RUN" mode Unit in mode "RUN" (fully automatic)
Pause	2.4 M	Unit in pause mode Unit in pause mode (semiAutomatic) Although the unit leads to the current step to get to the condition end, but enters HOLD mode. It allows the operator to forward or backward steps (step + 1 step - 1)
Hold	M 2.5	Unit in standby mode Unit mode HOLD (stopped) The production process is paused (controlled stop) Allows the operator to forward or backward steps (step + 1 step - 1)
EmHold	2.6 M	Free – It stops the unit in case of emergency The production process stops memoriimmediately (stop uncontrolled) tocurrently this mode is not implemented
Maint	2.7 M	Maintenance Unit mode Maintenance Possible only in step 0 It is possible to to the beginning of the program, not in automatic or Manual The unit is blocked
Gal	3.0 M	General alarm He isState of the unit is incorrect (watchdog, Hold requests),...)
GAlS	3.1 M	Stores General alarm General alarm was or is activated Signal must be reset por operator (OK + Reset) alarm)
SCE	3.2 M	State check error The unit is located in phase 2 (check of home) checked launch condition for the current program If the start condition is not fulfilled, there is no waterfalls of step
Watchdog	3.3 M	Alarm monitoring Control dand time for the current step Automatically turns on the alarm (GAl)
Step0	3.4 M	Unit in step 0 The unit is in the step 0
ReadyStart	3.5 M	The unit is ready to start The unit is ready for the next start of the program Number of prescription was delivered and recipe data has been read successfully from the database
UnitActive	3.6 M	Active unit (not step0). The unit is not in step 0
CIPModus	3.7 M	Unit in CIP mode for Visu The unit is in the mode of CIP (cleaning)
PA	5.0 M	Active step 1 = active step signal This signal is always 1 in the current step
PEH	M 5.1	Active step with retention of emergency function 1 = signal of Step Active and in the RUN mode, PAUSE or HOLD
PH	5.2 M	Active step with hold function 1 = signal of Step Active and mode RUN or pause Used for the control of the active components of plant
PP	5.3 M	Active step with hold function 1 = active step and run mode

Start	5.4 M	Unit start button The operator has pressed start the unit button
OperatorOK	5.5 M	operator OK unit button The operator has confirmed that the required manual action was performed
PFCycle	5.6 M	Step first cycle The step just activate. First cycle
PLCycle	5.7 M	last phase of the cycle The step It has reached the end condition and is disabled Last processing cycle
UnitNo	MB 10	Number of current unit
Phase	MB 11	Activated step
StepNo	MB 12	Number of the active step
Charge	28 MD	lot number Production lot number
PrId	MD 32	PrId ID unique production number
ProgNo	36 MD	Program number current (type number, type identifier)

17.3. Example of programming

Function Special in the processingmieprimary NTO

```
U "PFCycle"
S "Cnt". CNT [2]. Reset                resetcounter EO
```

Control of actuators

```
U "PH"
"Act" s. Act [12]. ACO                 Active el actedr in auto mode
"Act" s. Act [13]. ACO
"Act" s. Act [14]. ACO
"Act" s. Act [18]. ACO
```

After step on the condition

```
U "DIn". DIn [17]. GIS                 LSL
Or "Uxx". [3]. OK                  //Maximum time
= "PhaseEnd"                           End of step
```

Step with operator request

```
U "PH"
S "OpReq"

U "OperatorOK"
S "PhaseEnd"
```

Request for detention

```
U "Act". Act [12]. GAlS
Or "Cnt". CNT [2]. GAlS
Or "Watchdog"
```

S "HoldReq"

18. Options Home of the unit

The initial supply of variables is performed by the programmer (code system, planning systemcercania de production). Comenzado with the production then, these variables are passed a unit to another.

18.1. Structure of data

Assignment	Tipo	CpromoteIO
B24	BOOL	start option bit
B25	BOOL	start option bit
B26	BOOL	start option bit
B27	BOOL	start option bit
b28	BOOL	start option bit
B29	BOOL	start option bit
B30	BOOL	start option bit
B31	BOOL	start option bit
B16	BOOL	start option bit
B17	BOOL	start option bit
B18	BOOL	start option bit
B19	BOOL	start option bit
B20	BOOL	start option bit
B21	BOOL	start option bit
B22	BOOL	start option bit
B23	BOOL	start option bit
b08	BOOL	start option bit
b09	BOOL	start option bit
B10	BOOL	start option bit
B11	BOOL	start option bit
B12	BOOL	start option bit
B13	BOOL	start option bit
B14	BOOL	start option bit
B15	BOOL	start option bit
b00	BOOL	start option bit
B01	BOOL	start option bit
B02	BOOL	start option bit
b03	BOOL	start option bit
b04	BOOL	start option bit
b05	BOOL	start option bit
b06	BOOL	start option bit
B07	BOOL	start option bit
Val1	REAL	start option value
Val2	REAL	start option value
Val3	REAL	start option value
Val4	REAL	start option value
Val5	REAL	start option value
Val6	REAL	start option value
Val7	REAL	start option value

18.2. Example of programming

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

U "Uxx". StartOption.b02
= #Trub                        option 2 = Trub-adjustable
```

19. Data user

This area is reserved pARA user programming and It serves as a Clipboard of certain process or memory data for certain production processes. A statement (documentation) of the variables used always must be stored as a comment by bloce the network module step 1.

19.1. Structure user data

Assignment	TIpo	CommentIO
B24	BOOL	User bit
B25	BOOL	User bit
B26	BOOL	User bit
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	User bit
B18	BOOL	User bit
B19	BOOL	User bit
B20	BOOL	User bit
B21	BOOL	User bit
B22	BOOL	User bit
B23	BOOL	User bit
b08	BOOL	User bit
b09	BOOL	User bit
B10	BOOL	User bit
B11	BOOL	User bit
B12	BOOL	User bit
B13	BOOL	User bit
B14	BOOL	User bit
B15	BOOL	User bit
b00	BOOL	User bit
B01	BOOL	User bit
B02	BOOL	User bit
b03	BOOL	User bit
b04	BOOL	User bit
b05	BOOL	User bit
b06	BOOL	User bit
B07	BOOL	User bit
DINT0	DINT	user long int
DINT1	DINT	user long int
DINT2	DINT	user long int
DINT3	DINT	user long int
DINT4	DINT	user long int
DINT5	DINT	user long int
DINT6	DINT	user long int
DINT7	DINT	user long int

DINT8	DINT	user long int
DINT9	DINT	user long int
DINT10	DINT	user long int
DINT11	DINT	user long int
DINT12	DINT	user long int
DINT13	DINT	user long int
DINT14	DINT	user long int
Val0	REAL	user value
Val1	REAL	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

19.2. Example of programming special registration

Through the use of the transfer of the existing silos tien User Variables(e) a delay of a cycle. Like this, to enable a log entry in a silo can exchange, thereby the Elimination of the previous silo is registered.

Docu

UserDint 01: Memory of the next number of silo

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

L "SiloNo"
L "Dxx". User.Dint1
<>D
U)
L       0
&GT; D
)
S "ProtWrite"

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

20. Starting and stopping a sequncia (Run/Hold)

In some cases, the process does not must be stopped suddenly or not everyone will immediately be activated at the start. The reasons for this can be both economic, and prevent the lifting of current technologically.
In order to achieve an orderly shutdown and any, there are two possibilities. In simple contexts a delay to turn on or turn off time you can store directly in actuators, without programming. In more complex conditions, achieved a simple realization with variables of units of time.

20.1. Structure data

Símbolo	Operando	CoMentarIO
THold	REAL	unit of time in maintaining Time in seconds, the unit time Look forward
TRun	REAL	unit of time in "RUN" Time, in seconds, time of the unit in operation
TStepRun	REAL	Time in "RUN" of the step Time in seconds, While the current step is in "RUN"

20.2. Example of programming

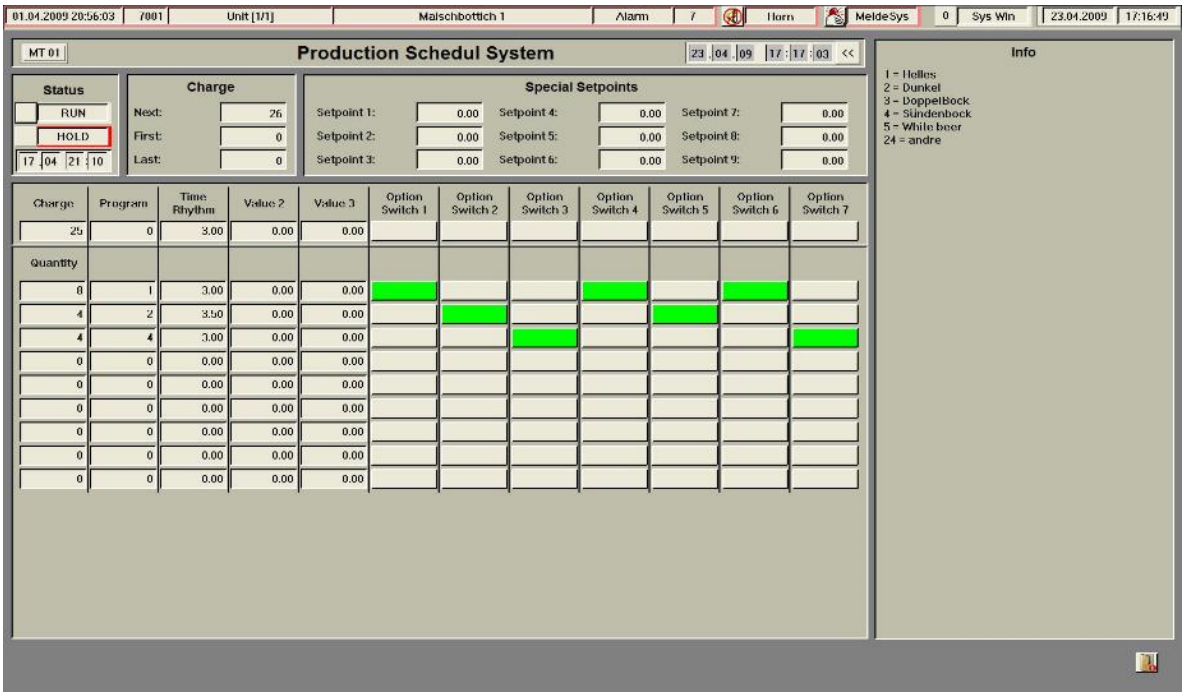
The following example shows a simple off orddesigned. The unit is in HOLD State, whether it is an event of a malfunctioning internal booking request, or the operator changes the unit to HOLD, by what time THold begins at funcionar. For example, actuators 12 and 13 are immediately stopped, delayed dthe actuator 18 to 10 seconds and delayed the actuator 19 to 15 seconds.

```
U "PH"
"Act" s. Act [12]. ACo
"Act" s. Act [13]. ACo

(L) "Uxx". U.THold
L 1. 000000e + 001              10 seconds
&LT; R
"Act" s. Act [18]. TOco         Delays on

(L) "Uxx". U.THold
L 1. 500000e + 001              15 seconds
&LT; R
"Act" s. Act [19]. ACo          Delays on
```

21. MSpecial odulos

21.1. Production planning system

It is a simple code integrated into the production system. Here the plan of production, number of productions is formed number of program (type specification), production rate and some startup options. How to use startup options is a thing of the programmer, The system in mode"Run", the lots are processed one after the other. The start of the respective unit is It takes place once it is free. With the start time (see State) can be changed from estado Hold by the way Run automatically.
The expected time for the production rate is passed a deferred batch.



The boot options are transferred automatically:

- Bits of the Production system -> Start Option Bits
- RITMO pRODUCTION of the System of Production -> Start Option Value 1
- Value 2 of the System of Production -> Start Option 2 Value
- Value 3 of the System of Production -> Start Option 3 Value

The Setpoint Special have adding setponit to a step. For example, a compensation of malta during la firsta cooking, or compensation for the last cooking volumes. The evaluation of the first or the last production takes place through the values of the batch. First batch cargua.

21.2. Example of programming

To use the production system, is called to the FC 281 and is transfer the corresponding unit number. Several units can be used as a starting unit, of course. In this case, the number of unit It must be passed in the form of a variable, that is the number of the corresponding unit. Do not call the block several times.

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