

# Analysis of the use of industrial control systems in simulators: State of the art and basic guidelines

Juan A. Carrasco\* Sebastián Dormido

*Trueba y Fernández, 12, 2B, 28016 Madrid, Spain*

(Received 2 March 2004; accepted 20 September 2005)

## Abstract

The use of industrial control systems in simulators facilitates the execution of engineering activities related with the installation and the optimization of the control systems in real plants. “Industrial control system” intends to be a valid term that would represent all the control systems which can be installed in an industrial plant, ranging from complex distributed control systems and SCADA packages to small single control devices. This paper summarizes the current alternatives for the development of simulators of industrial plants and presents an analysis of the process of integrating an industrial control system into a simulator, with the aim of helping in the installation of real control systems in simulators. © 2006 ISA—The Instrumentation, Systems, and Automation Society.

*Keywords:* Control; Control system; Simulation; Simulator

## 1. Introduction

The use of real control systems in simulators is not a new concept. It has been a hot topic in the power generating industry since the beginnings of the 1980s. The publications referred to that particular topic usually focus on the description of the simulated installations and on the description of the capabilities of the control system. This paper focuses on analyzing how an industrial control system can be integrated into a simulator.

It is important to notice that the decision of integrating a control system into a simulator is highly dependant of the simulation solutions provided by the control system supplier.

Nowadays, the use of a control system in a simulator is a simpler issue because of the technical characteristics of current process control systems and commercial process simulation tools.

Current technologies based on the use of open systems increase the viability of a control system optimization approach based on a previous installation of the same control system in a simulator. This integration into a simulator is easier nowadays due to the following reasons:

- The communication capabilities of both control systems and simulation tools have been highly improved and OPC for process controls (OPC) has actually become a standard [1–4].
- The control systems incorporate more tools related with simulation topics.
- The price of the necessary hardware is continuously decreasing and the cost of incorporating control systems into simulators is lower.

This mixed simulation solution (using real control systems) will be widely used in the near future, when the majority of the control systems will be based more on the use of software controllers. As an interesting example, ABB has been offering

\*E-mail address: jacarras@iies.es

during the last 4 years a software version of its distributed control systems (DCS) controllers based on the use of software controllers. The main characteristics of the ABB soft controller can be obtained from the ABB website [5].

This paper gives guidelines on how to perform a simulation that incorporates in an integrated manner a real control system. In Section 2 of the paper, the different alternatives for including real control systems in simulators are summarized. Section 3 deals with the main requisites of the process industry simulators. In order to avoid misunderstandings, each requisite is briefly described. A summary of the main simulation solutions provided by control system suppliers is presented in Section 4. The core of the paper, Section 5, presents an analysis of the possible behavior of the different components of a real control system in a simulated environment. In this section the authors present a small practical example related to this issue.

The conclusion section of the paper (Section 6) sums up the benefits of the aforementioned approach.

## 2. Control systems and simulators

The inclusion of a control system in a simulator can be carried out in different ways. In a great number of simulators a control system is included in order to provide a human machine interface similar to the interface of the actual physical plant

which is being simulated. If one of the aims of the simulator is the analysis of the control system response, then the control system can be included in a much more complete way. This means that there are different approaches for integrating a control system into a simulator. Unfortunately there are no standards for naming these alternatives in the simulation arena.

Fig. 1 shows a block diagram of the functional decomposition of a simulator of an industrial process. Obviously, the control system with all its selected components (the need for including each component depends on the selected integrating alternative) is an important module of the simulator. The models of the physical process of the plant, the instruments, and the actuators constitute the process simulation module. Simulation functionality represents the part of the simulator (it can be an independent software module) that handles the commands that manage the simulation session. Finally, in all the simulators used for training there is a set of facilities that helps the instructor lead each simulation session. When the simulator is not used for training, this last module can be called an interface for the management of the simulation session (by the final user).

The ISA standard for the construction of a simulator of fossil plants [6] specifies the alternatives for developing the control system module of a simulator. This standard mentions:

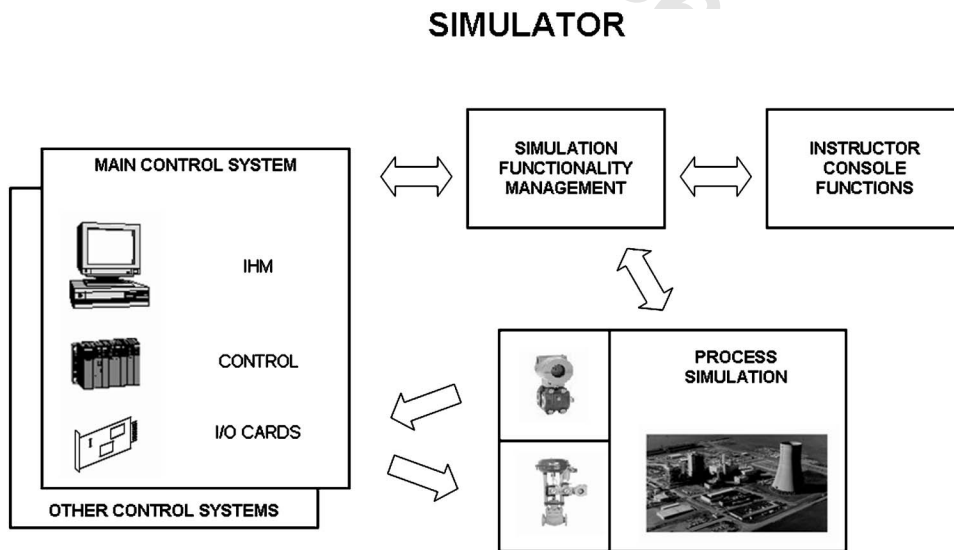


Fig. 1. Block diagram of a simulator containing real control systems.

Table 1  
Characteristics of control system simulation alternatives.

Characteristic	Emulation/simulation	Partial stimulation	Full stimulation	Virtual stimulation
Direct use of control algorithms	No	No	Yes	Yes
Direct use of operator graphics	No	Yes	Yes	Yes
Support of DCS engineering functions	No	No	Yes	Yes
DCS hardware needed	No	HMI	HMI and controllers ( <i>I/O</i> normally not used)	HMI?
Faster than real time	Yes	Yes (although trends and historical graphs can be affected)	No	Yes (although trends and historical graphs can be affected)
Initial hardware (HW) cost	Low	Medium	High	Low-Medium
HW maintenance cost	Low	Medium	High	Low-Medium
Cost for maintaining the simulator updated	High	Medium	Low	Low

- simulation, which uses alternate hardware and software programmed to emulate the instrumentation system's man machine interface without necessarily replicating all its functions;
- partial simulation, which uses the system hardware and software to replicate the man machine interface. However, some functions are emulated in the simulation computer (i.e., control loops, efficiency calculations); and
- full stimulation, which uses the system hardware and software modified to function in the simulator environment.

The aforementioned standard (1993) does not cover all the current options. The use of a simulation tool or the use of soft control versions are approaches that are not covered by the standard.

Up to now, the terms used for describing simulation approaches were not widely accepted. Nowadays, there seems to be a wider consensus with the following names:

- emulation/simulation (when the control system is not included);

- stimulation;
- partial stimulation [when only human machine interface (HMI) is included];
- full stimulation (when the control system is installed in the same way as it is in the plant); and
- virtual stimulation (when the control system simulation is developed with simulation tools provided by the control system supplier).

Table 1, completed from Ref. [7] sums up the main characteristics of the different simulation alternatives.

### 3. Basic simulation functions

The objectives of a simulator have a major influence in the simulation functions of a simulator. The main functionality characteristics that are usually present in all the simulators are:

- stopping the simulation (providing run and freeze states);

- saving and loading the current state and initial conditions;
- working in slow time and/or fast time;
- considering malfunctions; and
- replaying the simulation session (in certain simulators this functionality is known as replay).

Good details of these functions are presented in Ref. [8].

### 3.1. Stopping the simulation

The possibility of freezing the simulation without affecting the simulation response is needed in all the simulators. The determinism of the simulator can be influenced by the response of the simulator to the commands of start and stop.

When a control system is included in a simulator, the duration of the starting and the stopping process is a variable which should be optimized.

### 3.2. Saving and loading the current state and initial conditions

A previously saved state is usually known as simulation snapshot. The time needed for saving/loading an initial condition must be optimized. As in the response to run/freeze commands, the response of the simulator to the load of initial conditions can affect to the determinism of the simulator.

In certain simulators there is a possibility of going back through the simulation session and regenerating an initial condition from any previous situation. This functionality is usually known as *backtrack*.

It is important to reflect that the possibility of having access to the values of the variables archived in initial conditions could be very interesting for a final user, so a standard format of the saved conditions should be very helpful.

### 3.3. Working in slow time and/or fast time

The possibility of varying the simulation time is important for studies that are difficult to carry out in the plant because of their duration. The inclusion of a control system could affect this characteristic due to the fact that a control system is not prepared for working in “fast time” or in “slow time.”

### 3.4. Considering malfunctions

Considering malfunctions is an important characteristic in training simulators. Because of the robustness of actual control systems (with very long periods between failures) the malfunctions of the control system are not frequent. Any training on these failure situations (very infrequent in the plant) is very helpful. Usually, the architecture of the control system included in a simulator could be a simplification of the simulated plant control system in terms of redundancy. This simplification could hinder the nature of the malfunctions which can be considered in the simulator.

### 3.5. Replaying the simulation session

This characteristic is more difficult to be acquired in a simulator that is made-up by many different components.

## 4. Simulation aids provided by control system suppliers

The control systems simulation tools that are better known by final control system users are those which are directly provided by the control system vendors. The DCS have been the control equipment with a bigger relationship with simulation activities.

In the 1990s the suppliers that offered simulation solutions where the exceptions compared to nowadays where the suppliers that do not offer simulation solutions tend to be the exception.

- This section presents a summary of the main commercial simulation related tools provided by DCS suppliers.

The main alternatives provided in current commercial simulation tools are based on:

- a complete simulation solution where each control system is simulated on one computer and
- a simulation solution based on the use of software controllers that have implemented the same control algorithms that are being used in real plants by conventional controllers.

Fig. 2 summarizes the first of the two mentioned alternatives (the complete simulation solution). The figure has been improved from the Emerson professional operator simulation-based brochure.

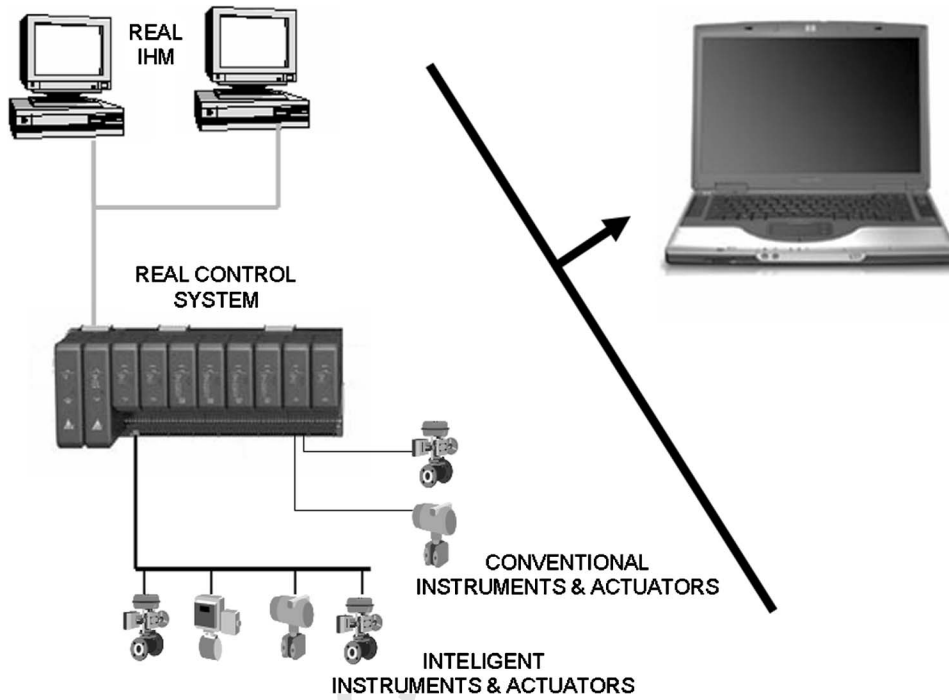


Fig. 2. Example of a complete simulation solution.

It stresses the possibility of installing the Training Delta V simulation solution in a single personal computer (PC).

Fig. 3, improved from Ref. [7], shows a comparison between the real control system installed

in a plant and a simulation solution based on software controllers. ABB is the supplier of both solutions (plant and simulator). This graph is an example of the second alternative.

Table 2 summarizes the main simulation alterna-

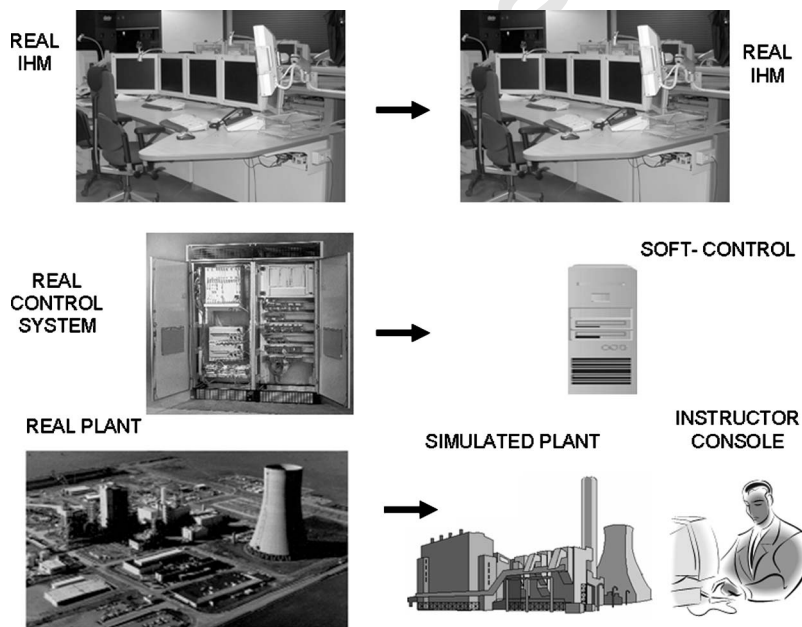


Fig. 3. Example of a simulation solution based on the use of software controllers.

Table 2  
Simulation tools offered by the main DCS suppliers.

DCS supplier	Name of simulation products	Based on software controller	Fieldbus simulation tools	OPC connection	Process simulation tool
ABB	Simpow Simcomx ITS	Yes	Fieldbus simulation cards	Yes	Simpow
Delta V Emerson	Delta V Simulate	No	Yes mimic (although it is connectable to other systems)	Yes	...
Foxboro Invensys	FSIM	No	...	Yes	Many Invensys products
Honeywell	Simulation environment, shadow plant	Not exactly	Fieldbus simulation	Yes	Shadow plant
Ovation Emerson	Virtual stimulation architecture	No	No	Yes	...
Mark VI (GE Industrial Systems)	Specific simulator for electrical plants	No	No	Yes	...
Siemens	SIMIT SIVAT	Yes	Profibus simulation cards	Yes	SIMIT
Triconex	TRISIM	No	...	Yes	Invensys has many products

tives offered by important DCS suppliers. The contents have been obtained from Ref. [7] and from the websites of the suppliers.

### 5. Including a control system in a simulator: Guidelines

The incorporation of a real control system in a simulator is an engineering project, so it must follow a typical engineering methodology. A good incorporation methodology must contain, at least, the following phases:

- project specification and conceptual design of the simulator;
- detailed design;
- hardware and software acquisition;
- project development;
- integration;
- simulator supply;

- installation and start-up;
- verification and validation;
- simulator maintenance; and
- training.

The use of some guidelines, covering the project specification (PS), detailed design (DD), project development (PD), and verification validation (V&V) can provide a good simplification of the mentioned methodology. The reading of these guidelines illustrate, easily, the incorporation project:

- analysis of the simulation solutions of the control system supplier (PS);
- selection of the simulator architecture, and how the control systems will be considered (PS);
- defining the needs of the simulation functionality (PS);



- analysis of the control system components (DD);
- implementation procedure (PD); and
- validation of results against the original plant (V&V).

Previous sections of this paper provide help in the project specification related guidelines. This section goes deeply into the “analysis of the control systems components” guideline and it finishes with a very simple example. A good reference of the implementation of a real control system in a big simulator is presented in Ref. [9].

### 5.1. Analysis of the control system components

This section presents an analysis of the difficulties encountered during the integration of a control system in a simulator. For this purpose an independent analysis of the different components of a control system should be carried out.

The main components considered in the integration of a real control system into a simulator are:

- inputs to and outputs from the process;
- basic analog control strategies;
- logic control strategies and control programs;
- advanced control strategies and operation support systems;
- human machine interface:
- process displays;
- alarm systems; and
- historical data management.

It must be stressed that the way of working of the control algorithms is the main issue to be analyzed. In the vast majority of the cases the rest of the components work in the simulator exactly the same way as they do in the plant.

#### 5.1.1. Inputs and outputs to the process

The way of considering the inputs and outputs of the control system could condition the possibility of using the same control strategies in the plant. The inclusion of input and output cards of the control system in the simulator depends on the characteristics of the control system.

The data filtering functions have a dynamic response which must be considered in the generating process and in the loading process of initial conditions. If no handling of the inputs/outputs has

been planned, it will be necessary to wait after any loading of initial conditions (the impasse needed for reaching a new steady state period of the transient which is caused by this initial condition loading).

Another alternative is the possibility of disconnecting the filtering actions during the loading of initial conditions. If the inputs and outputs are fieldbus based, the use of a fieldbus simulator could be an interesting option, but there are not many alternatives.

Mimic tool (from Munger) is a good simulator but its use is not extended outside Delta V simulation tools. The description of this tool is available at the Munger and Emerson Process Management websites, [10,11].

If the fieldbus is used for automatic control actions, the special considerations presented in the following sections must be taken into account for the integration of the inputs and outputs of the control system.

#### 5.1.2. Basic analog control strategies

The basic analog control strategies are usually performed by proportional plus integral plus derivative control algorithm (PIDs) and feedforward algorithms.

The main relevant topics related with the basic control algorithms of a control system which must be properly considered in the installation of the control system in a simulator are the following:

- assurance of a correct run-freeze transition;
- loading/saving initial conditions;
- similarities between simulator and plant time responses; and
- performance guarantee in different execution time modes (fast/slow time).

##### 5.1.2.1. Assurance of a correct run-freeze transition.

The easiest way in order to allow a run-freeze transition is the use of the auto mode of the control algorithms, which implies the use of the auto feature for run mode and the manual feature for freeze mode. Obviously, this solution is needed when the control algorithms do not have an initializing/hold mode. The choice is subjected to the implementations made for the control algorithms by the control system vendor. In some control loops the transitions from manual to auto state resets some values of internal variables in order to

allow a bumpless transition. The impact of this reset action must be evaluated because if a freeze command is made in the middle of a transient, the determinism of the simulator could be lost.

Of course, in simulator run mode some of the control loops of a plant can be in manual status, therefore a buffer that stores the current status of each control algorithm must be enabled. The physical position of this buffer (located in the machines in charge of making the simulation of the process or located inside the control system) depends on the characteristics of the control system.

Finally, it is also important to notice that some algorithms (a lead-lag for example) may not have a manual mode, therefore a different solution must be taken into account. The reduction of the engineering limits of the output of the algorithm (as small as possible in order to avoid any changes in the output), could be one possible solution. Another possibility lies in augmenting as much as possible the time scan in order to avoid any iteration before the transition to a run mode.

The action of putting all the loops in auto or manual mode after a run/freeze command must be synchronized, so the time needed for that action must be very small (not bigger than one scan period).

The use of this simple solution (“manual-auto transitions”) becomes complicated when the algorithms have activated their tracking parameter. If the tracking functionality must be maintained in the simulator, new buffers will be needed in order to save the value of the tracking parameter and the target value for the setpoint. Without these buffers the target value for the setpoint will be lost in a manual transition.

*5.1.2.2. Loading/saving initial conditions.* When the control algorithms do not have an initializing mode, the easiest way of implementing the loading of an initial condition is to update the parameters of the control algorithm. This update should be made placing the algorithm in manual mode and after the parameter change the correct saved mode of the algorithm should be restored.

A brief analysis for the PID algorithm is presented next. A similar analysis should be performed for the other control algorithms.

Although the PID algorithm is always digitally

implemented, in most of the documentation regarding control systems it appears in Laplace form (normally with few details about its final implementation).

Usually, it is implemented in the control system as an incremental algorithm using a structure similar to

$$\text{output}_T = \text{output}_{T-1} + \Delta\text{output}_T,$$

$$\Delta\text{output}_T = \Delta P + \Delta I + \Delta D,$$

$$\Delta\text{output}_T = K\Delta\text{error} + I:\text{error} \Delta t + \Delta D.$$

The use of this iterative type of algorithms in PIDs implementation, changes the control output of the algorithm into its main state variable.

Important additional state variables which must be initialized during the loading of an initial condition are the auto/manual mode and the setpoint value. Other state variables to be considered in a PID algorithm are the values of the control action parameters, the type of PID, the algorithm alarms, etc.

*5.1.2.3. Similarities between simulator and plant time responses.* If the simulator is going to be used in process optimizations and in control system optimizations, a detailed analysis of the time response of the control loops in the simulator will be very useful.

This time response analysis is essential for any possible extrapolation of the results obtained in the simulator. The main causes for differences between simulator and plant time responses are: the different kinds of inputs and outputs, deficiencies in the model of physical sensors/actuators, and possible delays in the communications in the simulator between the control system and the simulation models.

The less easily identified cause is the appearance of delays in the communications between the control system and the simulator. The following figures show the effect of a delay of 1 s in this communication in a real simulator.

Fig. 4 shows a similar variation in the setpoint in two loops, one without communication delays and the other with a 1 s delay. Although the control algorithms have the same parameters, the resulting time response is completely different. The figure shows “in red” or “with a dark line” the time response in the loop with delays and “in green” or “with a light line” the response in the



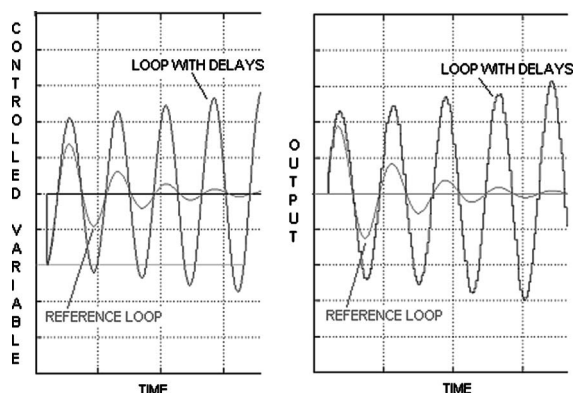


Fig. 4. Result in a simulation of a transient in a loop with delays in communication.

loop without delays. The time response of the process value is presented on the left side and the time response of the control actions is presented on the right side. The differences in the process value response are originated by the effect of the very small differences that appear in the control output of a loop due to the delays. In the first part of the transients the values of the control outputs are very similar, but the small differences observed in the loop with delays lead to an unstable situation.

Obviously, all the characteristics of the loop are affected by the delay (not only the step response).

The correction of the effect of the mentioned delays can be solved using theoretical or empirical solutions.

The best theoretical solution is the use of a Smith predictor. This predictor should have a delay parameter of the same magnitude as the communications delay. The Smith predictor can be implemented in modern systems as explained in Ref. [12].

The best empirical solution is the use of a trial and error approach. In this solution an interactive tool can be very helpful. The authors of this paper have developed an interactive tool for this purpose using Sysquake by Calerga [13].

An additional empirical solution that is very difficult to obtain is the use of formulas for transforming the initial PID parameters into new parameters which are able to overcome the delays. The authors have studied the effect of delays using a full battery of industrial plants (based on the battery presented by Astrom in Ref. [14]). They have obtained an example of correction param-

eters for first order plants (with and without integration) but their conclusion is that there are not easy expressions due to the high number of variables that must be taken into account). The correction needed for the PID parameters depends on the delay, control adjustments, time constant and type of plant. Their work in that area is available upon request.

Fig. 5 shows the results with one of the obtained formulas in a real simulator. The figure shows the time response of two control loops, one without delays and the other with delays and control parameter correction.

*5.1.2.4. Performance guarantee in different execution time modes (fast/slow time).* The most basic way of producing a variation of the execution time is to modify the scan time of the algorithm, but all the algorithms of a control system are usually prepared to work in real time and the modification of the time scan parameter only affects the output of the algorithm between two consecutive algorithm calculations.

A studied variation of the parameters of the control algorithms provides the possibility of working in different execution time modes.

If the relationship between normal time and slow time is defined as  $X$ , the corrections which guarantee slow execution time mode are:

- $I_{\text{SLOW TIME}} = I_{\text{NORMAL TIME}} / X$ ,
- $D_{\text{SLOW TIME}} = D_{\text{NORMAL TIME}} / X$ ,

where  $I$  is the integral action and  $D$  is the derivative action. It is important to notice that the proportional parameter of a PID is not affected by a variation of the execution time. The impact of the scan time of the algorithm could originate some differences between a transient in normal time and the same transient in slow time with the proposed correction.

The study becomes more complex when the algorithms are working close to their saturation zones. In this kind of transients the proposed linear correction is not perfect due to the nonlinear conversions that should be required to obtain an identical response in slow and in normal time.

A similar conversion should be made for guaranteeing fast execution time mode but it is important to notice that most of the commercial control systems are not prepared to work in fast time.

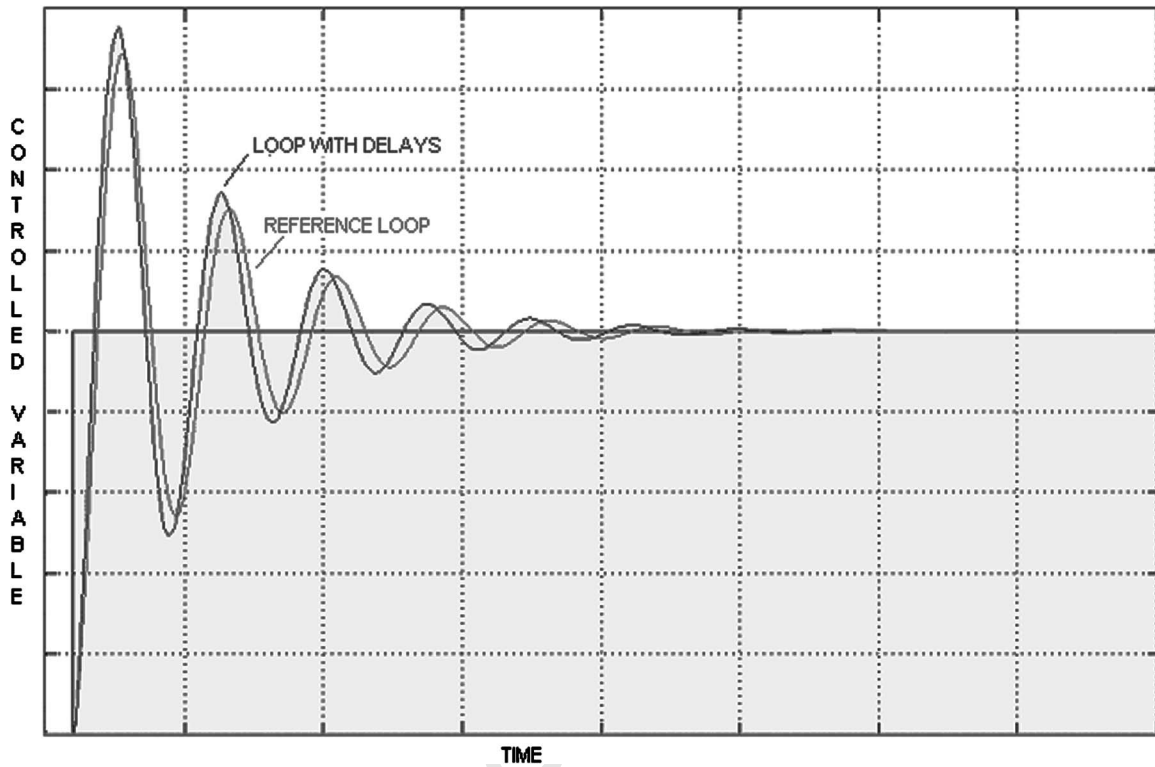


Fig. 5. Dynamic response of the delay correction vs response without delays.

### 5.1.3. Logic control strategies and control programs

In the majority of the cases the logic control strategies and control programs can work in the simulator without any variation. Their working scans (usually smaller than or equal to 1 s) contribute to this.

In any case, their status variables should be identified in order to know if any corrective action should be planned (similarly to the actions indicated in the analysis of analog control strategies). The main relevant topics that must be properly considered are the same that have been commented for analog control algorithms: Assurance of a correct run-freeze transition, loading/saving initial conditions, similarities between simulator and plant time responses, and performance guarantee in different execution time modes.

### 5.1.4. Advanced control algorithms and operation support systems

When an advanced control system is incorporated into a simulator the state variables of each of their advanced control algorithms must be identi-

fied. If this identification is possible (the documentation usually does not help on this matter) and a read/write access is possible, the incorporation in the simulator is easy, saving and restoring the values of this state variables in an adequate way.

Because of the enormous differences in advanced control systems each advanced control system should have its own specific handling.

It is important to notice that in many cases optimization of the control and the necessity of assuring a good response to the simulation functionality (run-freeze, load-save of initial conditions, slow-fast time, ...) is not an essential matter. Therefore, the integration efforts should be made in accordance with the needs of the simulator because the simplest integration (without taking into account any simulation functionality) could be an acceptable solution in cost-benefit terms.

Finally, it is important to notice that the control system can be complemented with operation support systems (also known as computerized operation support systems). The handling of these additional systems (that are usually based on the advanced control algorithms mentioned above)

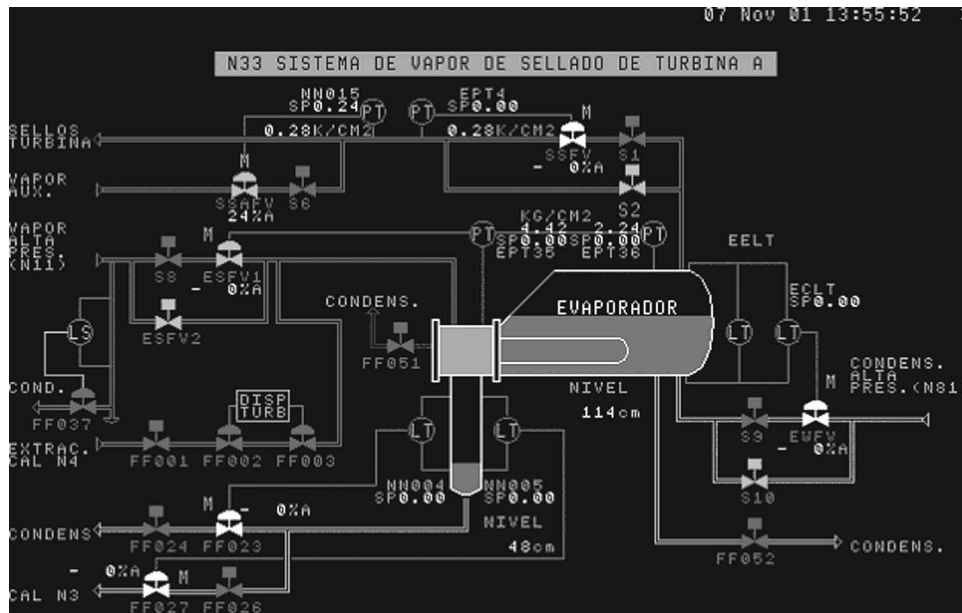


Fig. 6. Example of a simple process display included in a simulator.

can be carried out by following the suggestions presented previously. Advanced alarm systems and computerized procedure systems are two important examples of these types of systems.

#### 5.1.5. Human machine interface

The human machine interface is usually the most important topic in the integration of a control system into a simulator. The correct functioning of the human machine interface is usually used as the best indicator of the performance of the simulator.

In modern control systems, the human machine interface consists of “process displays” (mimics of the process), “alarm system displays,” and “historical data management displays.”

*5.1.5.1. Process displays.* The most important contribution of the use in the simulator of a control system is the possibility of using the same human machine interface that is used in the reference plant.

In older control systems the use of the same human machine interface required the use of the controllers. In modern control systems this use is not required.

Fig. 6 shows a very simple process display of a control system included in a simulator (it repre-

sents a small plant process system in detail). It is the same plant process display, and the effort needed for the integration into the simulator is minimal.

The handling of the alarms and trends that appear in the process displays should be consistent with the handling of the alarm system and of the historical data management system which are commented below. It is important to notice that the handling of real time trends is usually more difficult than the handling of historical ones because real time objects usually do not require the use of files with a known format.

*5.1.5.2. Alarm system displays.* The main difficulty for including an alarm system in a simulator is its consideration as a black box for the rest of components of the control system. Normally the operations allowed by the alarm system are scarce and they focus on an interface with the operators of the control system.

In addition, there is not a standard that normalize the characteristics of an alarm subsystem. Therefore the integration of an alarm system must be analyzed in a specific way each time that a control system is integrated into a simulator. However, the communication standard OPC is changing this situation. The alarm section in this standard presents a new way of providing an external

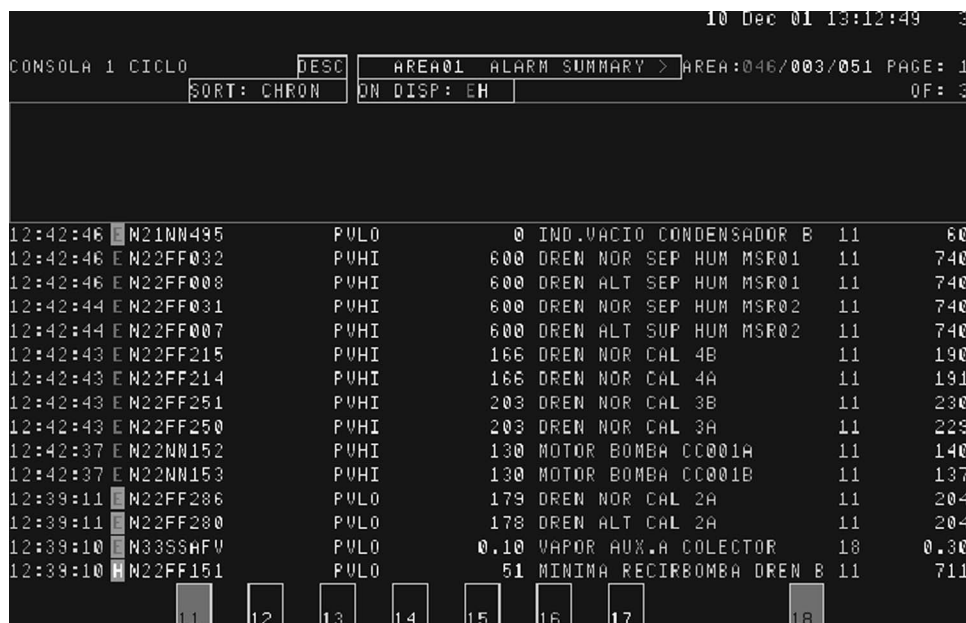


Fig. 7. Example of an alarm system display.

communication with the alarm subsystem of the control system. The use of this standard facilitates the integration of a control system into a simulator. Unfortunately, the alarm section of the OPC standard is the less implemented OPC subsection in commercial control systems, but this deficiency is likely to be corrected in a short time.

Fig. 7 shows the typical aspect of the alarm system summary display.

The main difficulties in the integration of an alarm system into a simulator are related with the handling of:

- the acknowledgement of the alarms;
- the dead band of the alarms;
- the rate of change alarm; and
- the handling of high level alarms.

The deadband which can exist in each alarm complicates the handling requiring a study of the evolution of the alarms in time. For the majority of the simulators this time analysis is not necessary and the deadband of the alarms can be disregarded. This disregard could cause apparent differences in the alarm system status after the different loadings of the same initial condition.

If the status of the alarm system is permanently recorded, the status of each alarm could be regenerated after the loading of an initial condition. The development of a specific program in

charge of coping with this situation can be avoided if a simpler solution is adopted. This simpler solution could consist in a complete reset of the alarm system, acknowledging all the alarms.

*5.1.5.3. Historical data management displays.* The main difficulties for the integration of an historical data management system are similar to the difficulties of paper registers in old simulators.

The historical archive system normally uses the clock of the system. The only solution to avoid the inconveniences provoked by this clock reference, is based on changing the time of the system in order to eliminate gaps at the end of each freeze state period and changing the time when a back-track action is made.

Fig. 8 shows the effect mentioned when no correction action is taken (left side), the effect when a simple correction consisting in deactivating the data collection during a freeze state (center subfigure), and the effect when a full correction is made (right side). Naturally, when the full correction is performed there is no problem. The setpoint is represented as a single line, the control action is represented with small rectangles and the process variable is represented with small vertical lines.

The effect mentioned above appears also in the real time trends that are included in the process displays. These trends are usually independent



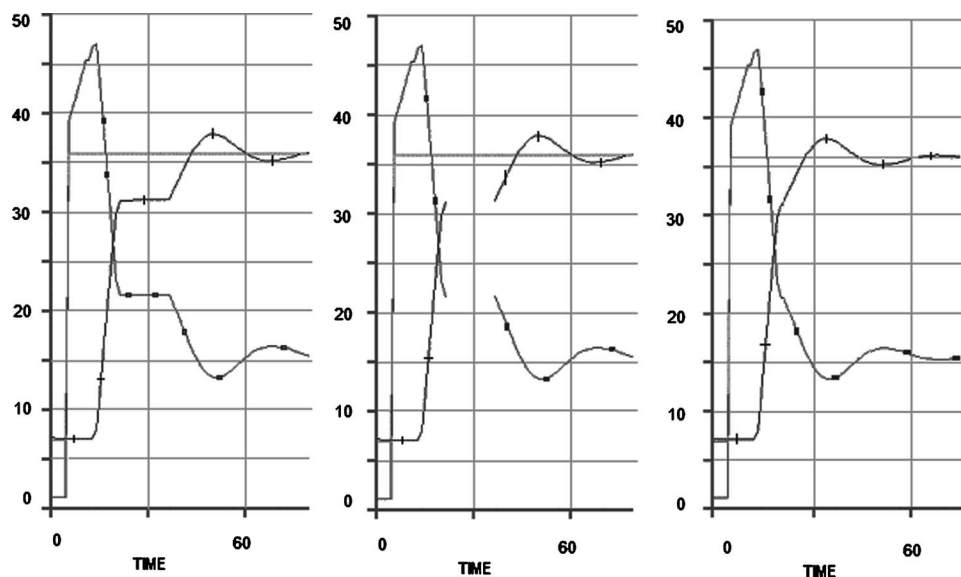


Fig. 8. Transient with a freeze state period (without and with correction).

objects/controls with a black-box behavior. In full correction cases the trend object should be substituted by a specifically developed object. It is important to point out that any modification of an object of the process displays implies the need to maintain different displays in the simulator from the plant displays.

Working in fast/slow time presents additional problems for the integration of an historical archiving system in a simulator. The time in the axes of each graph/trend does not correspond with the real simulation time (which can be bigger/smaller than the graph time).

When the historical archive system allows the use of an external clock, synchronization between the historical archive system time and the simulation time is possible.

The easiest solution for the integration of an historical data management system could be the same solution that was adopted in the majority of the old simulators which had paper registers, i.e., “do nothing” solution. This do nothing solution is optimal for training simulators because the historical register stores all the scenarios and all the actions of the trainees, although the effect of the freeze states can increase the difficulty of the analysis tasks.

### 5.2. Small practical example

This section presents a small practical example of the inclusion of a control system in a simulator.

The original control system is a system where the control functions are fully carried out by a commercial SCADA package (iFix from Intellution). The main characteristics of the control system are:

- Process—A tank of water with two valves, one for filling the tank and the other for emptying it;
- Alarms—One low level alarm (10% of tank level) and one high level alarm (50% of tank level);
- Basic analog control strategies—One Single PID algorithm developed in the SCADA database;
- HMI—produced by the SCADA package; and
- Input/output equipment—I/O analog cards connected through an OPC connection.

The practical application of the guidelines mentioned in this paper is presented below.

#### 5.2.1. Analysis of the simulation solutions of the control system supplier

The iFix from Intellution does not have any simulation aid so this analysis is obvious.

#### 5.2.2. Selection of the simulator architecture and how the control systems will be considered

The control system will be integrated into the simulator as a replica of the control system of the



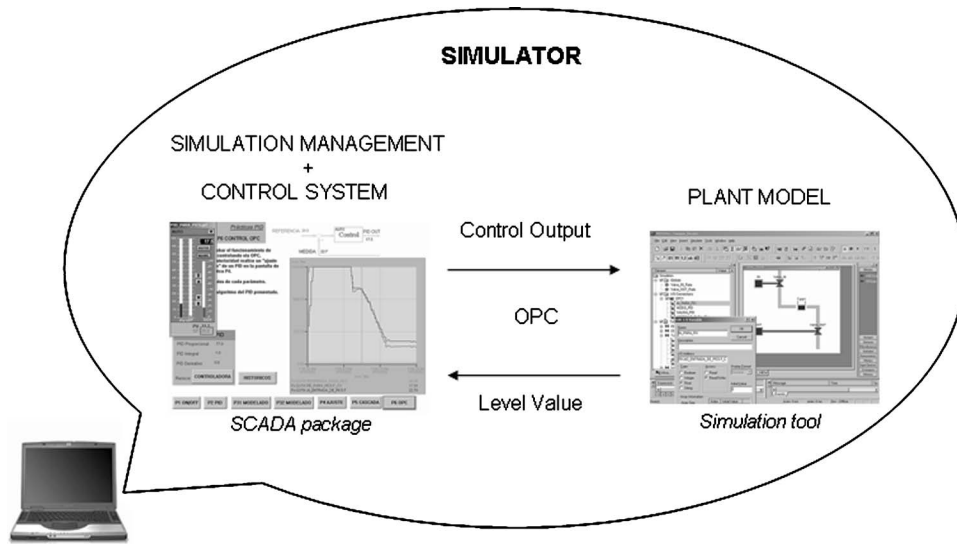


Fig. 9. Architecture of the simulator developed in the example.

plant, with the exception of the *I/O* system. Consequently, a “full stimulation solution: HMI plus control strategies” alternative has been selected. In this special case the HMI and the control equipment are included in the same software tool and they share the same hardware equipment: one single PC.

The simulation of the process will be developed using PicsPro from SST. The connection between the control system and the simulation of the process will be carried out using an OPC connection. The simulator management functions will be implemented inside the control system. The entire simulation will be installed in one single PC. The architecture is summarized in Fig. 9.

### 5.2.3. Defining the needs of the simulation functionality

The simulator will be used for demo purposes so the basic functionality needed is minimum. The simulator functionality specifications are:

- stopping the simulation (providing run and freeze states): yes;
- saving and loading the current state and initial conditions: Only one initial condition will be needed and it should act as a reset of the simulation;
- working in slow time and/or fast time: not needed;
- considering malfunctions: not needed; and

- replaying the simulation session: not needed.

### 5.2.4. Analysis of the control system components

The components that need to be considered are:

- inputs and outputs to the process;
- basic analog control strategies;
- logic control strategies and control programs; and
- advanced control strategies and operator support systems;
- human machine interface:
  - process displays;
  - alarm systems; and
  - historical data management.

**5.2.4.1. Inputs and outputs to the process.** The inputs and outputs will not be included in the simulator and the selected way of communicating the control system and the simulator is through an OPC interface. Therefore a change in the address of the input/output blocks of the database should be carried out. The input block of the SCADA database should be readdressed to the correct process simulation variable and the output block of the SCADA database should be readdressed to the variable that simulates the valve actuator. Fig. 10

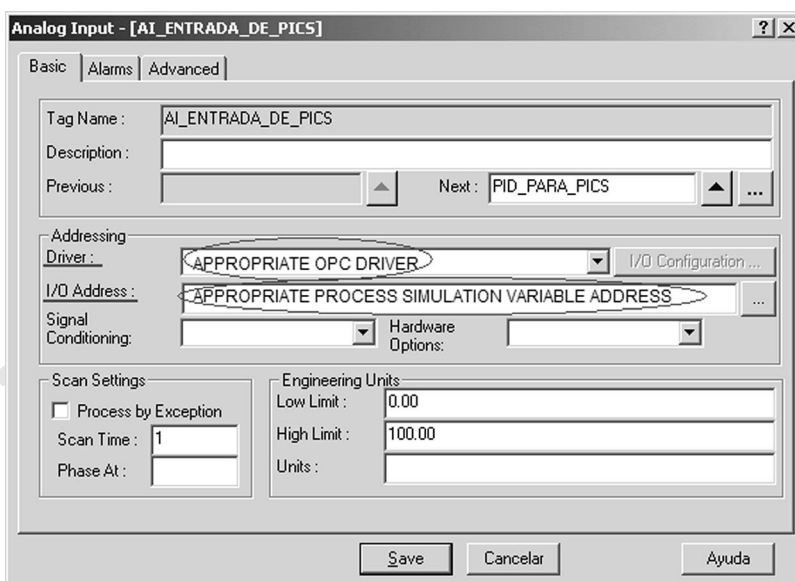


Fig. 10. Main configuration screen of an analog input block in iFix database.

shows the aspect of a database input block in the iFix SCADA. The driver related fields have been underlined.

5.2.4.2. *Basic analog control strategies.* As mentioned previously, the aspects that should be evaluated for the PID control algorithm are:

- assurance of a correct run-freeze transition;
- loading/saving initial conditions;
- similarities between simulator and plant time responses; and
- performance guarantee in different execution time modes (fast/slow time).

The control system will be installed as it works in the plants and the run-freeze transition will be obtained varying the auto-manual mode of the PID algorithm. In order to assure a correct run-freeze transition a correction like the one presented in the previous section is needed (the example shown in the previous section reflects the real Intellution PID behavior).

Only one initial condition will be implemented. The simulator will be able to load this unique initial condition but it will not be able to save new initial conditions. This initial condition will change the PID to manual mode and the control output will take the value of zero. The level will take the value of 50% of the capacity of the tank and the two valves of the process are then

closed. It is important to notice that the save/load action affects control system variables and affects process variables so some relationship should be established between the modules in charge of each part.

The implementation of the simulation functionality task will be carried out using buttons in the main process screen. The buttons will contain VBA code with access to the control system database. Because of the characteristic of the developed model, the simulation commands can reside inside the control system.

The communication characteristics of the connection in the simulator are very similar to the characteristics in the real plant. Consequently, no differences will appear due to the characteristics of the example implemented in the plant and the simulator. In this example, we can affirm that the simulator has a similar response to the plant response.

The simple example chosen will not work in slow time neither in fast time.

5.2.4.3. *Logic control strategies and control programs.* The real control system does not implement any logic control strategy or control program. Therefore no further considerations are needed.



Fig. 11. Main process display of the commented control system.

5.2.4.4. *Advanced control strategies and operator support systems.* The real control system has not implemented any advanced control strategies or any operator support systems. Once again, no further considerations are needed regarding this particular topic.

5.2.4.5. *Human machine interface.* As mentioned earlier, the main parts of the human machine interface of a control system are:

- process displays;
- alarm systems; and
- historical data management.

5.2.4.6. *Process displays.* There is only one synoptic in this example and the same display used in the plant can be used in the simulator. The unique change needed is the addition of two buttons, one for changing from run to freeze state (and vice versa) and other for Loading initial conditions.

Fig. 11 is an example of the plant process display.

5.2.4.7. *Alarm systems.* The simulator and the plant use the same alarm system which is the basic SCADA alarm system. Therefore, no change is needed. Fig. 12 shows the screen of the alarm con-

figuration section of an input database block.

This screen reflects the status of the alarm enabling parameter, the priority of the alarms, the dead band, and the values of the low and high alarm. Lowlow and Highhigh alarms will not be used (having the limit values of the signal range).

The only solution needed is the implementation of a total alarm acknowledgement when an initial condition is established.

5.2.4.8. *Historical data management.* No consideration is needed for the historical data management system. This decision will originate the appearance of flat lines in the historical data during freeze states.

### 5.2.5. Implementation procedure

The implementation procedure consists on carrying out the decisions mentioned in previous sections.

### 5.2.6. Validation of results against the original plant

In this demo example no validation has taken place, but it is important to remark the existence of this phase due to its possible importance in other simulators.

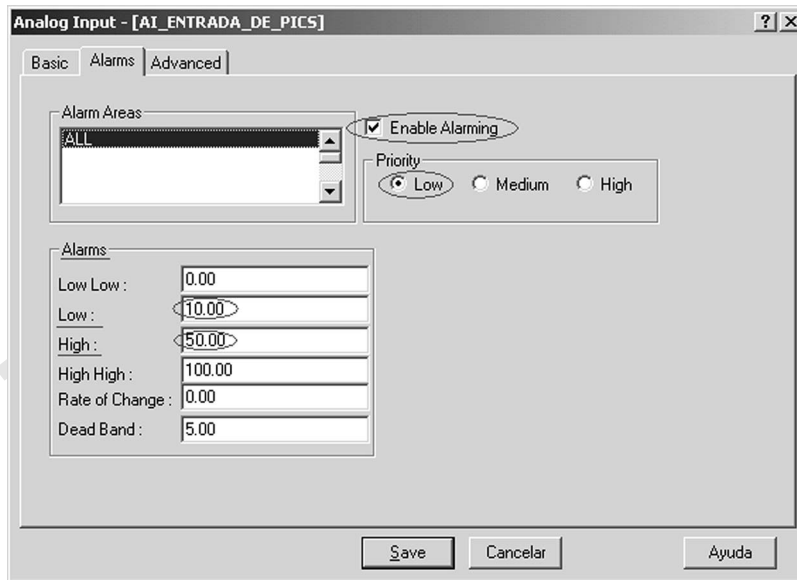


Fig. 12. Alarm configuration screen of an analog input block in iFix database.

## 6. Conclusions

The main conclusion that can be obtained from the analysis presented in this paper is that any control system can be included in a simulator. Of course, the integration of a control system in a simulator is highly conditioned by the existence of appropriate simulation tools.

When the incorporation is performed previous to the installation of the control system in the plant, it provides important benefits for the optimization of the control system and even the optimization of the design of the process plant itself.

The integration of the control system into a simulator offers the possibility of thoroughly checking all the software functions that will be used in the final plant. Because of the difficulty in assuring nonerror software, this approach has an added value for safety systems.

The present paper provides a simple way of including a control system into a simulator. This approach can be defined as a KISS approach (the traditional keep it simple stupid approach that is recommended in many engineering fields).

It is important to reflect that the experience of the authors indicates that the advantages of including control systems in simulators outbalance the disadvantages.

The main advantages are the following:

- The man machine interface can be fully de-

signed and validated previously to a plant installation. Physical aspects such as the quality of the screens can also be tested. The simulator is the best place for carrying out human factor studies. These studies are gaining an increasing degree of importance.

- The performance of the control system in the simulator can be similar to that in the real plant. Control loops and control programs can be fully tested in the simulator.
- The final user of the system can improve his confidence in the system when he sees the performance of the control system under extreme operation conditions in the simulator. It is not usual to see a control system working in extreme conditions in a real plant.
- It is important to notice that the use of the real system in a simulator allows a further independence from the control system provider than if a closed simulation solution is used.

Of course, there also are some disadvantages. These disadvantages are:

- There is a cost when including real equipment in a simulator.
- The maintenance costs of the simulator are increased if a real control system is included, and the maintenance staff will need a deep knowledge of control systems.
- The communication characteristics of the con-

trol system (mainly in DCS with an older design) could limit the rate of the data exchange. These limits may demand specific studies of time response. Fortunately, the control systems are continuously increasing their communication potentials.

- Fast simulation is not easy to obtain with real control systems.

Summarizing, the experience obtained by the authors indicates that the uses of real control systems in a simulation environment previous to the installation of the system in the real plant allows:

- an improvement of the design, development and validation of the HMI;
- an improvement of the design, development, and validation of the control programs and strategies; and
- an improvement of the adjustment of process control loops and control programs.

## References

- [1] Larsson, M., Modeling and control in Matlab for ABB's control builder. Lund Institute of Technology, Department of Automatic Control (2000).
- [2] Peršin, S. and Tovornik, B., Using OPC for the process control in matlab environment. University of Maribor, Faculty of Electrical Engineering and Computer Science (2000).
- [3] Pyykko, J. and Leppakoski, J., A novel training and engineering simulation concept for power plants. ISA Product Code: TP01POW500 (2001).
- [4] Ruppel, F. and Wysor, W., Future applications of simulators in process control. ISA Product Code: TP97ISA1105 (1997).
- [5] ABB Product Description Pages: Soft Controller, <http://www.abb.com>.
- [6] Fossil fuel power plant simulators-functional requirements. ANSI/ISA 72.20 (1993).
- [7] Thayer, T., Smith, K., and Brownlee, B., Trends in control system vendor solutions for simulator applications. SimSci-Esscor, ABB (2004).
- [8] ANSI/ANS 3.5-1998, Nuclear power plant simulators for use in operator training.
- [9] Carrasco, A. and Fernández, L., Control system modernization using simulator as an engineering help. Storejfell Enlarged Halden Programme Group Meeting, OECD Halden Reactor Project (2002).
- [10] Munger-Mynah Product Description Pages: Mimic, <http://www.munger.com>.
- [11] Emerson Process Management Product Description Pages: Delta V, <http://www.emersonprocess.com>.
- [12] Massimiliano, V., Performance improvement of Smith predictor through automatic computation of dead time. Yokowaga Italy, Industrial Automation Department (2003).
- [13] Calerga Product Description Pages: Sysquake, <http://www.calerga.com>.
- [14] Astrom, K. and Hagglund, T., *PID Controllers: Theory, Design and Tuning*, 2nd ed. Instrument Society of America, (1995).