БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ • BULGARIAN ACADEMY OF SCIENCES

ПРОБЛЕМИ НА ТЕХНИЧЕСКАТА КИБЕРНЕТИКА И РОБОТИКАТА, **59**
PROBLEMS OF ENGINEERING CYBERNETICS AND ROBOTICS, **59**

София • 2008 • Sofia

State-of-the-Art and Development Trends in Intelligent Data Acquisition and Distributed Control for Complex Industrial Processes

Nick Ivanescu

University Politehnica of Bucharest, 313, Spl. Independentei, Bucharest, Romania
E-mail: nik@cimr.pub.ro

1. Process control nowadays

The stage of automatic process control in Romanian industry can be discussed from two points of view:

1) the situation of older and updated objectives;

2) the situation of new objectives.

Between older and updated objectives you can find factories, chemical plants, oil refineries, that maintained the technology and improved the process control system. Here also processes can be classified in several categories: continues, analogue and batch ones. Most of the improvements can be found especially in the oil industry, in refineries, bulk plants, oil chemistry, one of the reason being that more money are destined for automation in these fields.

In order to obtain performances regarding the management of such an objective it has to start from a concept of hierarchical control situated on levels of competences and responsibilities like in Fig. 1.

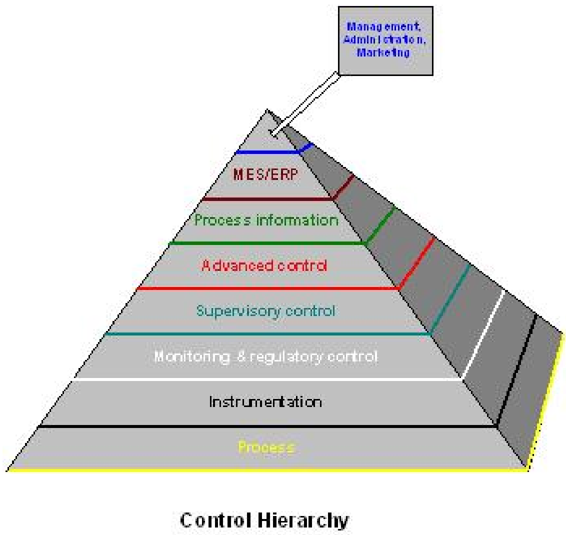

Fig. 1. Control hierarchy

A Distributed Control System (DCS) with centralizing of all data and operations is the most adequate and most efficient to manage such objective, fulfilling the first four levels of the pyramid. A corresponding architecture of this concept will be presented in continuation (Fig. 2).

In elaboration and designing of a supervising system, especially for those with an increased degree of hazard, the problem of safe process control divides in different activities: Actual control of the process, where the only performance criterion is to keep the manufacturing process stable and in optimal conditions; to achieve this, a last generation DCS provides classical PID and advanced Fuzzy algorithms, acquisition cards and analogical commands with 16 bits resolution. On the other hand, for optimizing and obtaining spectacular results, above the classical control system should be implemented an advanced one able to online adjust the regulation parameters of the controllers, based on process information taken from intermediary points, essential in stabilizing the quality of the final product. In this respect a series of real-time analyzers have to be installed in the places where:

- the quality of the final product is determined;
- intervention is possible such that correction measures can be taken.

One of the newest and performant analyzers, based on Nuclear Magnetic Resonance (NMR), together with the mathematical model of evolution of components and classical laboratory analysis, is able to supply precise and real time information about the composition of the product. Due to this fact the advanced control system can automatically adjust the references of the classical control system such as to avoid incorrect composition of final product. As an example an operating screen is shown in Fig. 3.

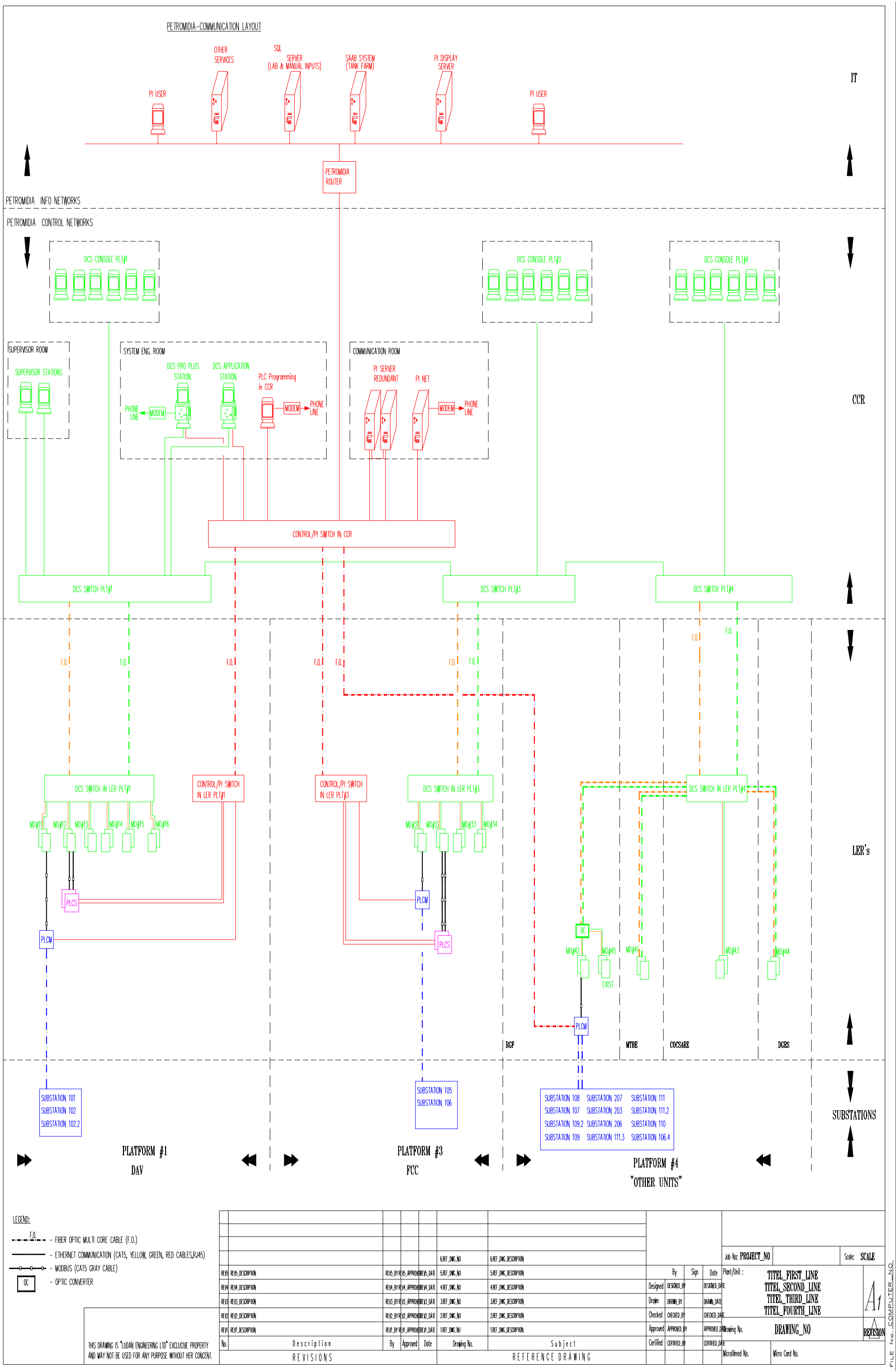

Fig. 2. Distributed control system

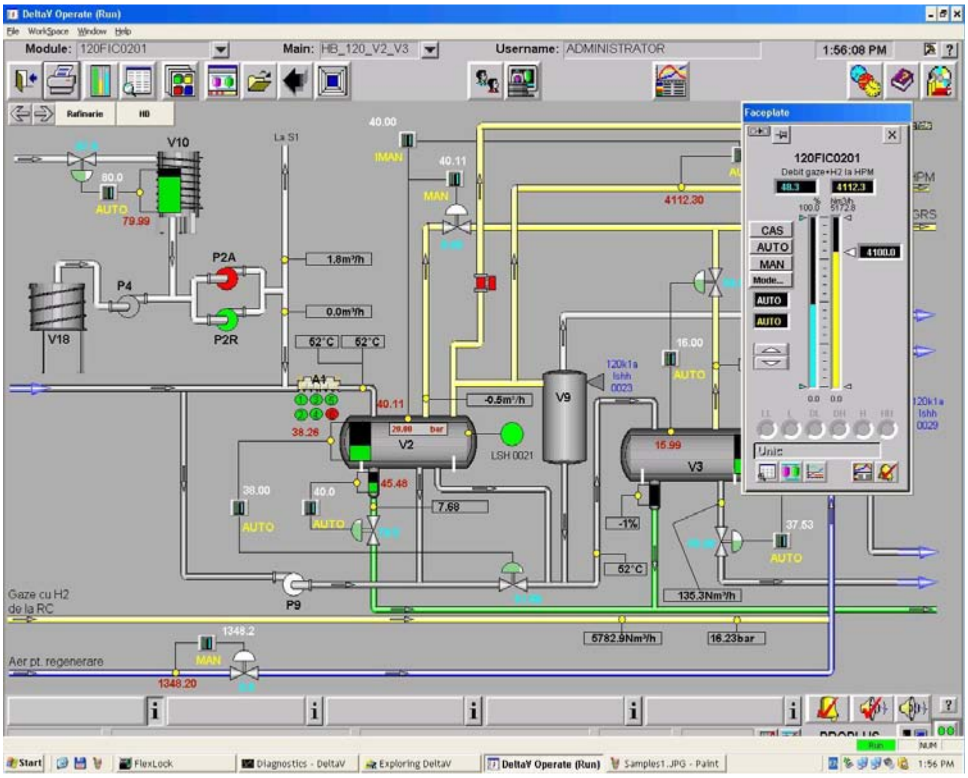

Fig. 3. Example of an operating screen

For a dynamical view of the system and for taking some predictive decisions the possibility of watching the timing evolution of parameters' behavior is absolutely necessary. This is also useful for viewing the system response to perturbations and to diagnose it. The system provides visualization and data storage mechanisms running simultaneously with the control tasks such as the response time is not affected. Also every modification of a parameter or overflowing of a limit must be memorized in order to allow a detailed diagnose.

Above the control system, the informational system is also necessary, it centralizes data from all the systems, inputs/outputs, raw materials/products utilities, laboratory, accountability and marketing systems, so that production efficiency structured reports and it's scheduling can be obtained as fast as possible, so that the enterprise as a whole can respond as efficiently as possible to market stimuli.

The factory's security represents an equally important issue that the automation has to accomplish; a point that is rarely seen but which probably has a higher importance than the production task. For this matter, the security standards impose safe systems with a very high redundancy degree, in order to assure that, no matter what the circumstances, the system responds in order to avoid catastrophes. For this case, the designed emergency shutdown in safe conditions systems ESD (Emergency Shut Down) are completely independent from the control systems and have to be classified according to a redundancy level, such as CPU level, power supply, inputs/outputs, two out of three, etc. Here we have the PLC systems who don't allow modifications through a user interface.

2. Current situation of automation in Romania

The new objectives' situation: generally the new objectives are represented by factories of small dimensions but usually of higher complexity. The onset of these usually involves using automated lines in which human intervention is minimal; this movement can be found especially in the alimentary and pharmaceutical industries. The technology producers provide complete lines with all the command and control systems that are based on the concept and implementation already presented. In cases where the system has smaller dimensions, as in the data transmission cases, the solutions implemented in grater numbers are based on PLC and SCADA interface. Also, in this new objectives situation, apart from the lines which we have already discussed, important automation components are the utilities that in certain cases present complexities comparable with the production system itself.

For example, in a dairy factory, the auxiliary systems can be:

1) steam necessary in the fabrication process;
2) production, fire fighting water;
3) ambient cooling and conditioning system;
4) power supply and protection system,

which, in the end, usually have an inputs/outputs number equal with the process part.

In chemical and petrochemical industry a front place is detained by DCS where big producers like Emerson and Honeywelle already supplied systems for big companies. As an example Rompetrol and Lukoil companies implemented the control system with Emersons' DeltaV System, while Petrom OMV used Honeywell' TDC3000 and Experion systems at their refineries because for reducing the costs they preferred to keep the existing cabling.

In the new installations smart transmitters and sensors connected to field buses (ProfiBus, ControlNET, DeviceNET) started to be used instead of classical transducers, have two big advantages:

– reduced cost of cabling;
– predictive maintaining.

The producers mentioned above are already supplying, for these type of smart transmitters, programs packages that give information about both the system as the transmitters (sensors, control valves) enabling in this way the possibilty to determine when a field device is not correctly functioning anymore.

Regarding the chemical industry, big plants like Azomures, Oltchim, Interagro, using a somehow smaller versions of the systems mentioned above, an updating of the control system was accomplished.

PLC producers are also powerfull presents in romanian industry:

– Allen Bradley;
– Siemens;
– Bosch;
– Kontron.

For example many middle-size applications were implemented with Siemens products. Lately a big israelian producer (TNUVA) opened one of the biggest and most modern factories (dairy produce) where the control system is based on Siemens 400 series together with Proleit system, a combination betwwen SCADA and DCS. DCS system from Siemens did not enter the Romanian market yet (as far as we know).

Another example is about a Romanian refinery that started the modernization of the control system five years ago and owing to the change of the old analogue system with a modern digital one, a spectacular improvement of the products' quality together with a significant lower price resulted.

The refinery is an assembly of large installations located on an extended area, installations that have a great degree of dependence between them and formatting together a system with an impressive number of measurement points, reaching 10,000-20,000, according to the size and number of the installations

Some other interesting applications are in the field of environment protection, for example some water treatment stations are controlled nowadays with Allen Bradley PLC's. In some cases data transmission was implemented with GPRS using miniplc's from Siemens together with Wincc SCADA software. In the thermic and electrical energy transportation domain, intelligent counters are connected on an M-Bus and communicate wireless with the base.

To correctly choose the main control device for any industrial system, several factors must be taken into consideration, such as:

– the number and type of input/output signals;
– the complexity of the software that must be developed;
– the communication facilities that are requested and can be fulfilled;
– the easiness of developing a human machine interface for supervising and command;
– the budget allocated to this part of the control system.

Modern devices are integrating high performance PC functionality, field buses and I/O modules. Some may even have a powerful Pentium processor, Ethernet interfaces, USB for keyboard and mouse, DVI interface for LCD or CRT displays and also can handle hundreds of I/O signals. Another big advantage of these devices is that even the human machine interface run on them, so practically you don't need another PC to supervise the process or to send manual commands to the controllers.

The software for these modern devices should have different functions, like:

– graphical user interfaces, for configuring the process, for monitoring all the production stages and all the factory areas, for manually command the installations in the factory;
– automatically accomplishment of the planned production;
– implementation of the constraints and of the inter-conditioning between different machines;
– sorting and mediation of analogue signals;
– network communication with PC's.

First let's briefly discuss some **theoretical** approaches regarding the conception of an automation software project. The easiest and the most natural type of programming languages that should be used for the control part of a program is Sequential Function Chart. This programming language is supported by the big majority of PLC's producers. Even if this language is not implemented in a PLC and other languages, like Ladder Diagram for example, must be used, there are algorithms that help the programmer to "convert" a logical diagram in a Ladder Diagram. On the other hand, other types of jobs, like implementation of inter-blocking, inter-conditioning of events can be done easier in a Ladder diagram type of program.

When several industrial processes had to be controlled by a single PLC, the best idea is to **separately** control every execution device (like motors, actuators, valves and so on) in a separate logical diagram. If several devices' related events are connected somehow, it's better to develop a single logical diagram for their control. Relationship or synchronization between logical diagrams is made by **global variables**.

Usually PLC does communicate with a pc by means of a serial port (RS232 or RS485) or of an Ethernet interface (usually available only in more expensive and newer PLC's). Different protocols are implemented in the PLC in order to communicate (like Profibus, TCP/Ip and others). Sometimes the programmer must develop his own communication protocol if none is implemented in the PLC.

Modern controllers have already installed an OPC server that can be used to exchange data between the PLC and other computers connected in the same network. This is a very useful feature because several PC's can communicate with the controller and can be used to monitor all the production process or even to send commands to the installations (if program allows this).

In the latest development programs the user can create as many as visual interface he needs and can run these interfaces directly from the PLC. Practically a PC is not needed anymore to supervise the control process. For this example one visual screen was developed, representing a specific area of the factory. In Fig. 1 you can see a final products storage area, with its complex transport systems, formed by conveyors, elevators and deviators. Buttons on the screen can be simply pushed with the mouse and will start/stop motors. Color of the objects usually indicates its status (started, stopped, full, empty and so on), according with the value of associated sensors.

3. Modern solution for controlling a feed plant

Nowadays, in former communist countries like Romania, almost all the factories built long time ago were privatized. But because of old technology existing in these factories, most of them needed to modernize and automate.

Our university was involved in several automation projects for some factories in the last decade. One of the recent projects we worked for was to design and execute a complete control system for a combined fodder producing plant. From the beginning it must be specified that we had to find the solutions to command and control the existing machines and the only things that could be replaced or added were some sensors and transducers.

Practically we had to accomplish three things:

1. Design of the command and control system, taking into account the electrical characteristics of the machines, the number and type of the electrical signals coming from and going to the system.

2. Development of a powerful software system such as to allow manual control of the whole plant, all-time visualization of all signals and commands and full automatic control of the production process.

3. Real-time communication with personal computers in a local network, allowing managers and other personnel to supervise the production flow and results.

3.1. Description of the plant

Combined fodder plant produces food for industrial grown poultry. This food is a mixture of different types of cereals combined with some concentrated products. Practically, there are three main areas inside the factory:

– Raw material storage area, consisting of several storage bins
– Weighing and mixing area, where practically the final product is obtained
– Finished product storage area, consisting of several storage bins, where the resulting product is stored waiting to be taken over to the poultry farms.

In Fig. 4 part of the plant is represented:

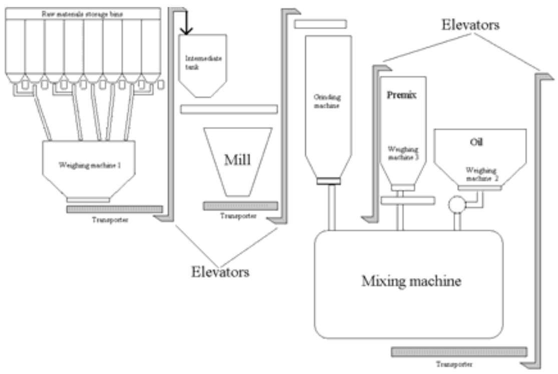

Fig. 4. Production part of the plant

Let's take a brief look at the components of the plant shown in Fig. 4:

- There are 8 raw materials storage bins, containing 8 different types of cereals.
- From these bins material can be extracted in the weighing machine 1, by means of several extractors.
- Weighing machine 1 is used to weigh specific quantities of cereals, with a maximum of 2000 kg supported by the machine. The machine supplies an analogical signal in tension, which we converted in a unified 4-20 mA signal.
- The intermediate tank between the weighing machine 1 and the mill is needed because the speed of the mill is lower than the evacuation speed from the weighing machine 1.
- The mill is used for milling the cereals.
- The grinding machine is used for grinding the milled cereals.

- The mixing machine is a large tank where the processed cereals are mixed together with some special oil and another component (premix).
- The materials circulate between the machines by means of several transporters and elevators.
- The weighing machines 2 and 3 weigh oil and respectively premix extracted in the milling machines. Details will be given later in the article.

The production flow can be briefly described as follows:

1. The human operator must establish the quantities for every type of cereal, for oil and for premix that must be part of the final farm-produce. Quantities are determined for only one charge of extraction, because the capacity of the weighing machines is limited. Also the number of charge is specified, in order to produce the whole amount of material wanted.
2. The process starts with the extraction of specified quantities of cereals into the weighing machine 1.
3. The weighing machine 1 is emptied into the intermediate tank, if some specific conditions are fulfilled. Immediately a new extraction begins, for the next charge.
4. The milling and grinding processes are done automatically, some sensors signal when the correspondent tanks are empty.
5. When the grinding is over, material must be transferred in the mixing machine. After a short time oil and premix are also unloaded in the mixing machine.
6. The mixing process lasts several minutes (configurable period of time), at the end of it the machine is opened and the finished product is transported to the finishes product storage bins and one charge could be considered ended.

It’s necessary to say that in order to obtain a great productivity, another charge must be in course of processing, even if the previous one is not finished. Of course the detailed process implies several constraints and inter-conditioning, some of them will be recalled later in the paper.

Before designing the hardware and software automation solution, it was necessary to make an inventory of all analogical and digital signals that must come from and go to the installations. After hard work 192 digital inputs, 126 digital outputs and 12 analogical signals were identified.

Because of large number of signals involved, we looked for a solution with only one programmable logic controller that should handle all these signals. We chose Kontron ThinkIO P from German company Kontron.

ThinkIO device is an innovative concept to integrate high performance PC functionality, field buses and I/O modules. It has a powerful Pentium processor, two Ethernet interfaces, USB for keyboard and mouse, DVI interface for lcd or crt displays. For automation applications it can run programs developed in 3S Software Codesys package. Also it can handle hundreds of I/O signals, so it fulfilled our necessities. Another big advantage of this device is that even the human machine interface runs on it, so practically you don’t need another PC to supervise the process or to send manual commands to it. We chose Linux operating system for this PLC.

Before designing the software solution, there was an important job to do, essential for correct results of the production process.

3.2. Standardizing the weighing machines

All three weighing machines have some tension sensors that output an electric signal (theoretically) proportional with the weight found in the machine. The weighing machines 1 and 3 have four sensors, mounted in the four corners of the weighing tank, that output a tension (0-100 MV) proportional with the forced laying on the respective sensor. The sum of the four tensions is converted in current, in the range of 0-20 mA, and is connected to an analogue input of the PLC.

A potential problem could appear if the sensors are not the same type (different output tension range) or are not mounted perfectly symmetrical. Unfortunately we faced such situation at the weighing machine 3. One of the sensors output a different tension range than the other three sensors.

The standardization procedure implies the determination of the relation between the weight of the weighing machines and the output current of the measurement sensors, practically is the relation between the weight and the engineering units resulting from the analogue-digital conversion inside the PLC.

To determine this relation, we used 100 standard 20 kg units. The weighing machines 1 and 3 can support up to 2000 kg load. To determine the specified relation, we loaded the machines with 200 following 200 kg, until the 2000 kg were reached. After that, units were taken away, 200 kg each time, until there was no load on the machine. Fortunately, the histerezis phenomenon was very small and practically did not matter. The characteristic of the machines is shown in Fig. 5, 1 is for cereals weighing machine, 2 for premix and 3 for oil weighing machine.

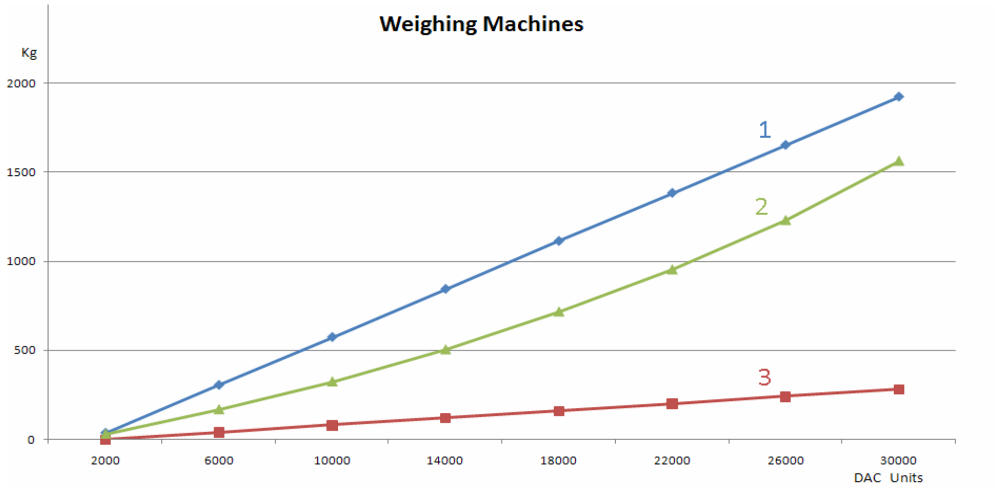

Fig. 5. Characteristic of the three weighing machines

Machine 1 proved to be linear so we adopted the following formula for calculating the instant weight value

(1) weight1 := (can_value – init_value1) / scale_1

where:

– **weight1** is the instant value of the weight, measured in kg;

– **can_value** is the numerical value obtained by converting the analogue signal (0-20 mA) on 12 bits;

– **init_value1** is the numerical value obtained from sensors when the machine 1 is empty (because the weighin machines have their own weight);

– **scale_1** is the approximated slope of the line, calculated using the arithmetic average of all can/kg ratios in all the measured points (see Fig. 2).

Machine 3 has a different behaviour because one of the sensors is a different type from the other three, outputting a tension in a smaller range than the other ones. The result is that the resulting graph is more like parabolic type. Anyway because to determine the parabola equation is very difficult, we choose the variant to linearize the graph on 10 portions. So the formula for one portion looks like:

(2) weight3 := (can_value – init_value3) / scale_3i

where:

– **weight3** is the instant value of the weight, measured in kg;

– **can_value** is the numerical value obtained by converting the analogue signal (0-20 mA) on 12 bits;

– **init_value3** is the numerical value obtained from sensors when the machine 1 is empty (because the weighing machines have their own weight);

– **scale_3i** is the approximated slope of the approximation line on portion **i**, calculated from the values of start point and end point of the respective portion (see Fig. 2).

Weighing machine 2 has only one tension sensor so the relation between the value in kilograms and the output tension is linear, so we used the same formula for all weighing range (0 -120 kg):

(3) weight_oil := (can_value – init_oil) / scale_oil,

where the elements of the equation have the similar significance as in previous equations.

3.3. Extraction process

Another delicate process that had to be correctly controlled is the cereals extraction and premix unloading. The cereals are transported from the storage bins with a system of conveyors and elevators and are loaded in the weighing machine 1 using several extractors. But because the end of the extractors is some distance above the weighing machine, there is still some material in the air that will fall down in the weighing machine, even after the extractor's motor is stopped.

The quantity falling down in the machine after stopping the extractor differs because of different factors:

– type of cereal;

– quantity wanted to be extracted;

– distance from the end of extractor until the weighing machine.

The same considerations apply for unloading the premix in the mixing machine, but the quantity falling, after the unloading lid is closed, is small.

The solution adopted in order to minimize the extraction errors was to use some estimated "in air" quantity so that to anticipate the quantity loaded in the machine after the extractor is stopped. In this case extractor should be stopped before the weighing machine measures the desired quantity. So the estimation (for one material) has an initial value but it is adjusted at every extraction with the simple formula

(4) $$\text{new\_estimation} = \text{old\_estimation} + + 0.5 \times (\text{last\_extracted\_quantity} - \text{wanted\_quantity});$$

where:

– **new_estimation** is the "in air" estimated quantity that will be used to determine the moment when the extractor must be stopped at current extraction;

– **old_estimation** is the estimation used at the previous extraction;

– **last_extracted_quantity** is the concrete quantity extract previously;

– **wanted_quantity** is the ideal quantity that should be extracted.

In this way in (3)-(4) extractions error goes below 1 kg practically. Taking into account that a complete production cycle has tens of extractions, the total extraction error is very small, less than 5%.

After the weighing machine 1 is loaded with all types of cereals needed for current charge, the machine is unloaded completely, if several conditions are fulfilled. Premix and oil quantities are measured somehow different, when they are unloaded from the weighing machines in the mixing machine, but the procedure is similar.

3.4. Software design

Codesys logical software package was used to develop the control programs. The software project should have had different functions, like:

– graphical user interfaces, for configuring the process, for monitoring all the production stages and all the factory areas, for manually command the installations in the factory;

– automatically accomplishment of the planned production;

– implementation of the constraints and of the inter-conditioning between different machines;

– sorting and mediation of analogue signals network communication with PC's.

The structure of the software project is shown in Fig. 6. The idea was to separately control almost every sub-process that take place during the production flow (extractions, mixing, unloading and so on) Most of the programs are written in Sequential Function Chart (SFC) programming language, the "*interblocking*" program is written in Ladder Diagram (LD) language and other smaller programs are written in Structure Text (ST) language. All of these programs run **simultaneously** in PLC's memory.

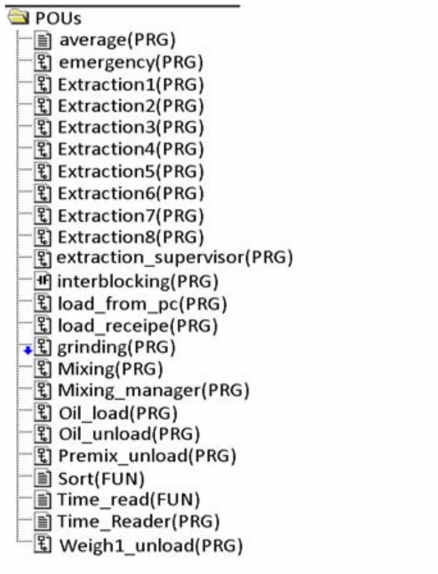

Fig. 6. Structure of the software project

Briefly described, the programs have the following functions:

– *extractioni* programs take care of the extraction of every type of cereals, one program for one type of cereal (written in SFC);

– *interblocking* program takes care of all inter-conditioning between different installations together with automatically stopping some motors when they reach their limits (see Fig. 4); for example one elevator cannot be started if a connected transported was not previously started; the program is written in LD language because it must check all the conditions at every plc cycle;

– *loadxxx* programs manage the communication between the PLC and PC (written in SFC);

– *Mixing* program controls the necessary timing needed for the mixing process of the materials;

– *Extraction_supervisor* program supervises the extractions programs. It decides if and when an extraction should start.

Other programs manage the loading and unloading of oil and premix, calculate the average values of analogue signals and so on.

The project can work in two modes practically:

– **manual** mode, where only specific commands can be given to some of the machines;

– **automatic** mode, when the production process is automatically managed by the PLC.

One of the major issues to be efficiently implemented was the correct management of **timing**. We chose a solution to use only one timer variable for all SFC type programs that is started when automatic mode is chosen. In all SFC programs that need timing comparisons, solution adopted was to read this **global timer** every time is necessary and to make the desired calculations of time passed. Other solution tested, using special timing functions, like Timer On (Off) Delay did not give expected results in SFC programs, but performed well in LD diagrams. When the automatic mode is over the global timer is stopped and reset, in order to avoid the situation when the timer could reach its maximum and overflow. Timer is not needed in manual mode.

Another special request was to minimize as much as possible the production time, in order to increase productivity. The solution chosen to achieve this goal was to extract the cereals immediately after the weighing machine 1 is unloaded in the transporting system. In this way no time is lost when the milling machine is prepared to receive material the weighing machine 1 is emptied without any delay.

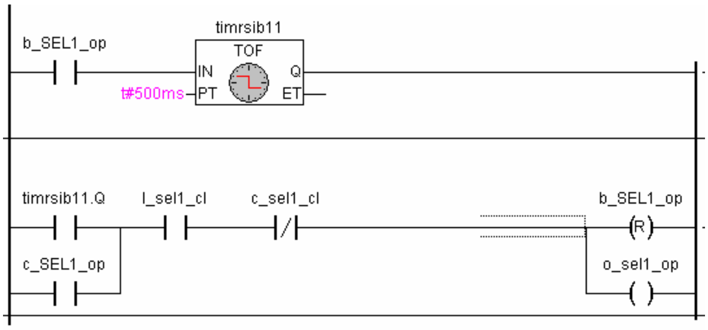

Fig. 7. Automatically stopping a motor in LD program

In Fig. 7 you can see an example of how Timer Off Delay function block is used in a LD program:

– because the impulse duration of variable *b_SEL1_op* (starts a motor rotating clockwise) is too short, it was extended with 500 ms.

In the second rung of the diagram you can see that the command is reset when:

– motor reached the corresponding limit;

– confirmation from counter-clockwise contactor is not active (failure situation).

In this way the operator must only start the movement of a motor and the controller automatically stop it when one of the conditions are fulfilled.

Another issue to be correctly managed was the emergency status. Because of the nature of the installations of the factory, there are frequent situations when the product process must be immediately interrupted (extractor clogging, transporters stuck and so on).

When an emergency appears (that could not be detected automatically) the operator must stop all the processes immediately by simply clicking a button, implemented in the user interface. Inside the programs, almost all the steps verify the status of emergency button and the status of several sensors (some failures or misbehavior is automatically detected). When emergency is detected all **SFC** programs stop in their current step. Depending on the gravity of the situation, the operator has the possibility to resume the process from the point it was stopped or to reset all the programs and start for the beginning (Fig. 8).

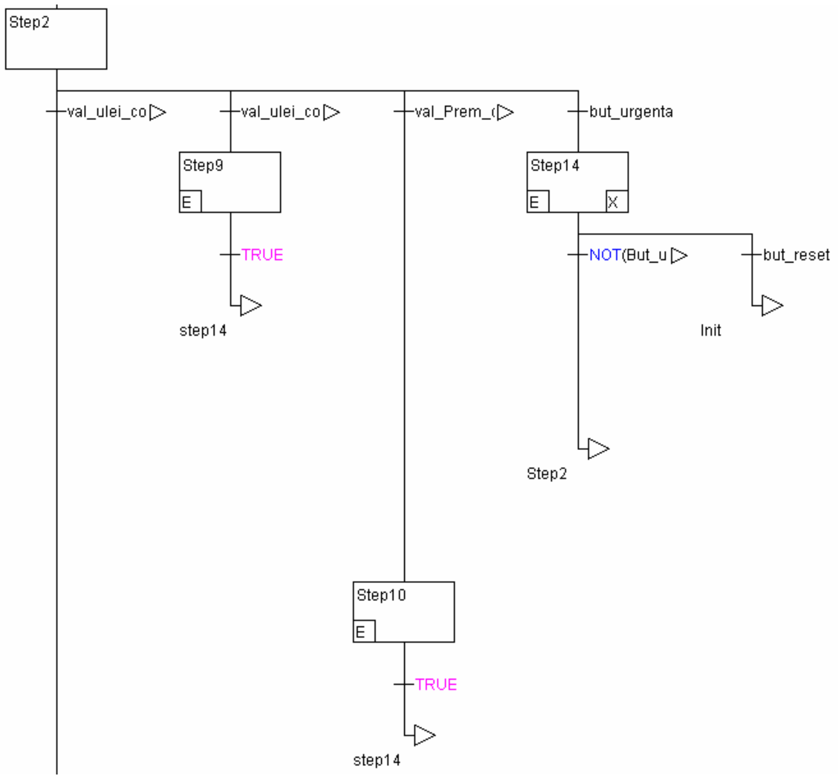

Fig. 8. Managing emergency situations in SFC

You can notice that from **Step2** has four conditions of transition, first one is the normal behaviour of the process, conditions two and three appear when some material is missing in its bin and the fourth one is true when the operator presses the emergency button on the interface screen. Last three transitions lead the program to an *"emergency"* **Step14**. From **Step14** the graph can evolve to **Step2** (resuming the program) or **Init** step, practically the program going to its initially status.

3.5. Communication part

ThinkIO controllers have already installed an OPC server that can be used to exchange data between the PLC and other computers connected in the same network. This is a very useful feature because several pc's can communicate with the controller and can be used to monitor all the production process, to configure the recipe or even to send commands to the installations (if program allows this).

At this moment one pc program was developed having the following behavior:

– the operator can edit, save, load recipes and send them to the plc on the network, acting as an OPC client;

– reports are printed at the printer after each charge finishes.

In Fig. 9 you can see the GUI of the PC program.

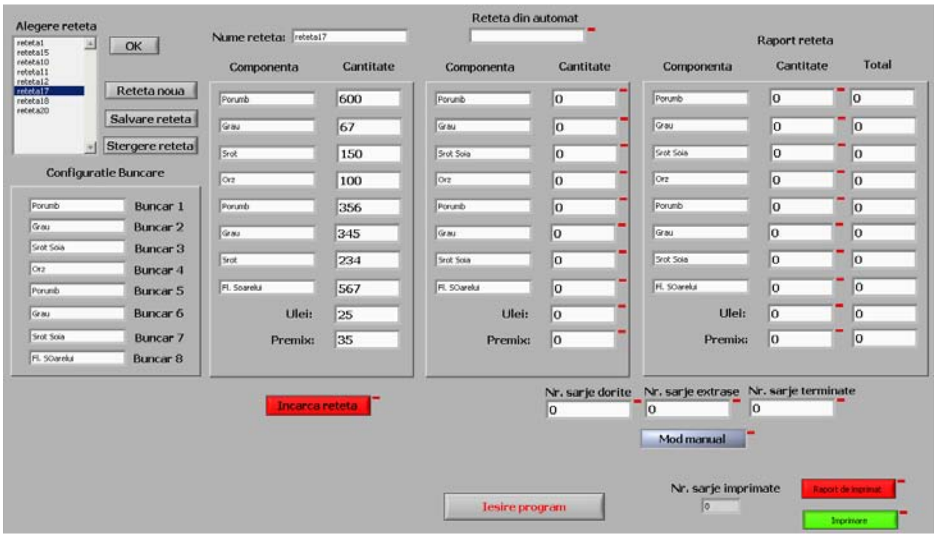

Fig. 9. Communication program from PC

3.6. Graphical User Interface on PLC

In Codesys development program the user can create as many as visual interface he needs and can run this interfaces directly from the ThinkIO PLC. Practically PC is not needed anymore to supervise the control process.

For this project we developed five visual screens, each one representing a specific area of the factory. For example in Fig. 10 you can see the final products storage area, with its complicated transport systems, formed by conveyors, elevators and deviators. Buttons on the screen can be simply pushed with the mouse and will start/stop some motors. Color of the objects usually indicates its status (started, stopped, full, empty and so on), according with the value of associated sensors.

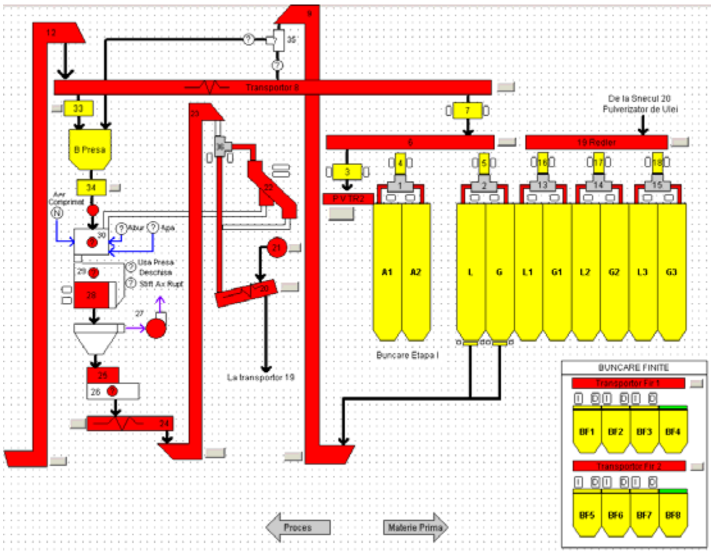

Fig. 10. Final products storage area

In Fig. 11 you can see the main area of the factory, where the production process takes place.

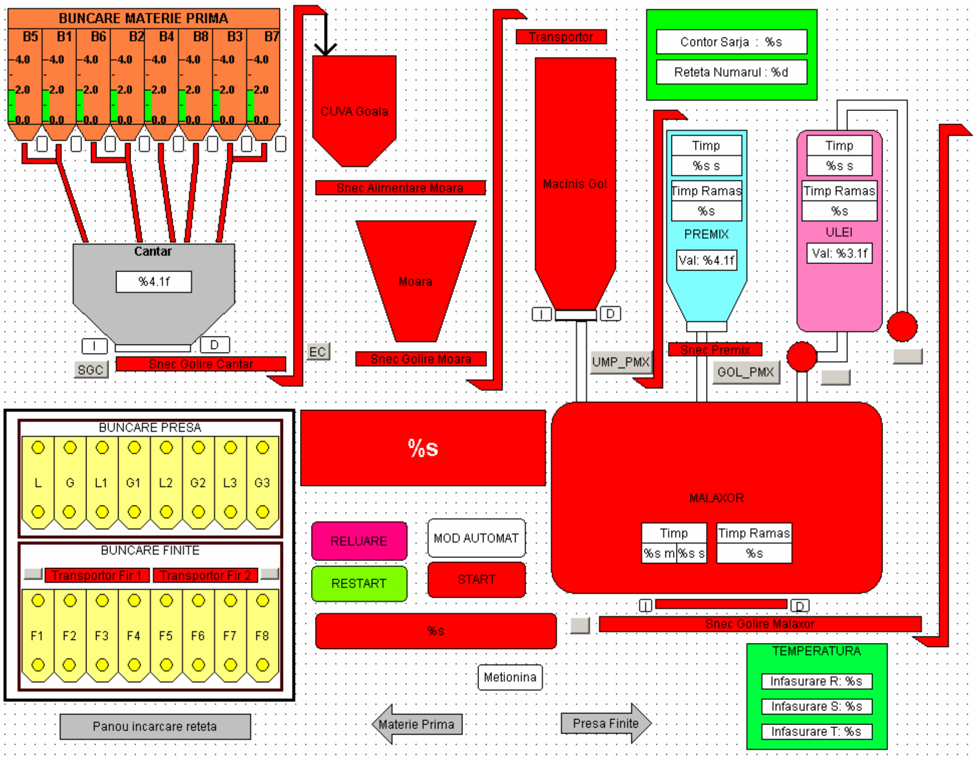

Fig. 11. Main production area

In this screen the operator can see information from all the sensors in the area, even the values from the weighing machines. Also operator can manually command the execution elements from this area; he can start automatic production or stop it by pressing the emergency button.

4. Results and conclusions

Regarding the first part of this article, the conclusion is that in Romania automation field is reviving nowadays. After a period of regression while many factories got closed, situation became better, rather large investments were made in building new factories or modernizing new ones. Many important producers of automation equipments and solutions opened branches here.

The only worrying thing is the lack of specialists in automation, many engineers left Romania for western countries. Also, although from automation universities hundreds of engineers graduate each year, few of them remain in the industrial filed, most preferring to find a job in the IT field.

Regarding the industrial application presented in the second part of the article, the first conclusion is that 2 important requests have been accomplished by the automation system:

1) the ratio between different components contained a final product must be very close to the programmed ones (maximum 5% error);

2) the quantity of final product must be also closed to the programmed total quantity (about 2-3 % error was requested).

After several weeks while the system was tested the results were like this:

– condition 1 was fulfilled, error was very small (2%);

– final quantity was about 5% more than the desired quantity, because of estimated extraction, some small losses in transporting system and so on.

The solutions taken was to reduce a little the scales for calculating the kg values at the weighing machines, until the resulted quantity was approximately 2% more or less the desired one. In that moment the automation could be declared finished, with very good results.

The solution chosen for this automated system is very modern and has several advantages upon, let's say, classical solution using other PLC's models:

– process control and supervising do not need a PC practically, all programs and visual interface run in PLC;

– fast and reliable network communication with PC's, by OPC server, different from widely used serial RS232 or RS485 communication in other controllers;

– small working cycle due to fast processor;

– extremely large number of I/O modules that can be managed (hundreds of digital signals already monitored) by a single controller.

The future work that can be done is to make a supervising user interface program that can access the controller directly from internet, so even the owner of the factory can be informed about the production flow even if he is far away from the factory.

References

1. I v a n e s c u, N., S. B r o t a c, T. B o r a n g i u. A Distributed and Configurable Architecture for Controlling Industrial Processes. Preprints of IMS2001, 2001, 132-137.
2. I v a n e s c u, N., S. B r o t a c. Supervising and Control of Industrial Processes Using Internet Technologies. Preprints of IMS2000, 2000, 78-82.

Состояние и тенденции развития интеллигентного сбора данных и распределенного управления при сложных промишленных процессов

Ник Иванеску

Политехнический университет, Букурещ
313, Spl. Independentei, Bucharest, Romania
E-mail: nik@cimr.pub.ro

(Р е з ю м е)

Работа делает обзор самых новых технологий и решений для управления сложными промышленнымш процессами на основе современной ситуации автоматизации в Румынии. Представлен и современный подход управления промышленными машинами и производственными процессами в пыщевой промышленности.