"ЕНЕРГЕТИКА І АВТОМАТИКА"

№ 4(34)

UDC 62-7

DEVELOPMENT OF THE FRAMEWORK FOR THE CONTROLLERS OF THE PROCESS BASE MANAGEMENT SYSTEM TO MEET THE REQUIREMENTS FOR INTEGRATION WITH OTHER SUBSYSTEMS AND TO IMPLEMENT SERVICE FUNCTIONS AND DIAGNOSTICS SERVICE

Alexander Pupena, Ph.D.

Roman Mirkevich, postgraduate student

Oleh Klymenko, Ph.D.

Volodymyr Polupan, postgraduate student

Department of integrated automated control systems

National university of food technologies, Kyiv, Ukraine

e-mail: pupena_san@ukr.net, roman.mirkevich@gmail.com, frank._.s@ukr.net, serunder@mail.ru

Abstract. *We consider approaches to develop a framework for application of software for programmable logic controllers as components of the base system process control (BPCS) with regard to their integration with other systems (SCADA, SIS, BPCS, MES / MOM) and increased requirements to the diagnostic process and service process and equipment.*

The concept of software framework is designed for controllers of PAC (Process automation controller), based on open standards IEC 61131 and is not used for producing specific solutions. The structure and the internal implementation of library elements of the frame is based on ISA-88, ISA-95, ISA-106 (Draft) ideas which provide the simplest integration of control systems into a single integrated automated control system.

The paper proposes mechanisms to improve the diagnosis of the process and the system, quickly setting up the system and restore its functionality after failures. The frame is a set of interrelated elements of library functions and function blocks, which are described at their interface level, interaction principles and algorithms of functioning. It is assumed that IEC 61131-3 languages will be used for the implementation of framework for the particular controller, but it is not a prerequisite. Set of frame elements can be expanded and supplemented with additional functionality without infringing the general principles.

First time proposed universal structure for process control of milk production by using modern standards for construction of integrated automated control systems.

Introduction. Worldwide to improve observability, controllability and safety operation of critical infrastructure is constantly developing, reviewing and improving regulations and standards. Specifically in the world accepted number of standards for objects and systems of increased danger, which are mainly based on the norms of ISA, in particular ISA84 (IEC 61511), ISO13849, IEC62061, which in many countries have adopted as binding. According to the accepted norms, such systems should include(Figure 1):

-two control subsystem such as basic process control system (BPCS) and safety instrumented system (SIS);

- human-machine interface via panel and SCADA/HMI;

- alarms and events management system (Advanced Alarm Application).
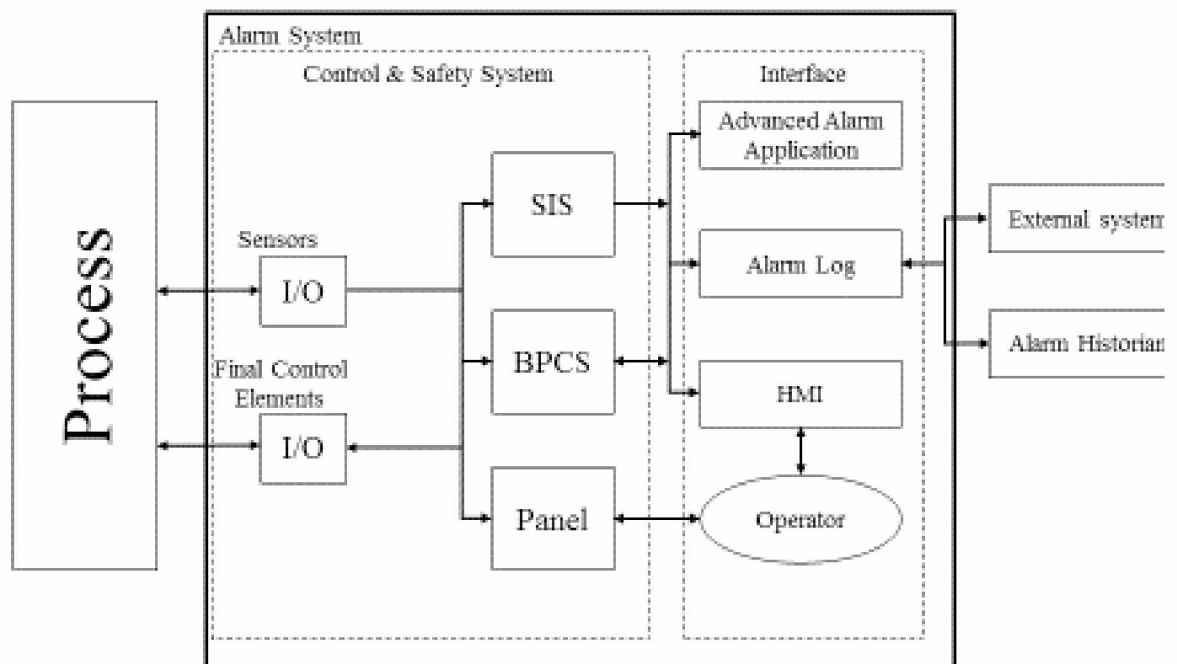

Figure 1. Alarm System Dataflow

Basic process control system provides the functions of process control and it is the central component, that is responsible for the integration with other subsystems.

Together with SCADA/HMI subsystem BPCS should implement all necessary production functionality, diagnosing process and system and service system. It must be possible to integrate with upper level management (MES / MOM).

Materials and methods. Among the most popular issues for this level of control, regardless of the object to date select the following:

- Low observation of object even when a sufficient amount of data;

- "Static" diagnostic process without reference to the type of product and conditions (typical for Batch-productions);

- Poor implementation of self-diagnosis and consideration of failures in the BPCS;

- Lack of awareness of the system of MES/MOM of false data;

- Insufficiently considered mechanism of functioning of alarms and events management system;

- Considerable time for manifestation fact malfunction and eliminate the causes;

- Significant spending resources on staff training.

A number of these problems resolved by applying the standard ISA 18.2 (Management of Alarm Systems for the Process Industries). One of the ISA committee considers the problem of low observability, who later plans to release standard ISA 101. To implement the ideas laid in the basis of the above standards required level of software and hardware. In practice, functional support, given in standards, is very different in concrete means SCADA/HMI. This leads to the fact that most of the functionality implemented independently, that the most simple and flexible to implement in BPCS, if it is implemented by a programmable controller.

Problems diagnostic process mainly related to static configuration alarms and events management system. Most of the important norms for process variables are set statically by development tool, what leading to significant problems reconfigure the system in the future. At best, these values are set by the operator by means of SCADA / HMI, but even so this approach is not suitable for variable time processes when set

point depends on the type of product and must be specified in the recipe. For these processes alarms and events management system and BPCS should function on principles of Batch-control, ie ISA-88.

Problems of integration PCS with the upper level (MES/MOM) usually associated with ontological inconsistency of these levels. Upper level systems require more information than just data values. Need additional information about the quality of the data, state process variables and equipment, state system, etc. That is for the quality of system operation and achieve maximum efficiency, the level PCS should be prepared for integration with the upper levels. Today the most well developed and proven integration mechanisms considered principles enshrined in the standard ISA-95. In turn, the requirements for PCS in accordance with concepts this standard imposed standards ISA-88 and ISA-106 (currently is under development).

All this leads to the need to develop basic control systems, taking into account the above requirements and using standard approaches. For systems where realization of basic process control system (BPCS) made on programmable controllers, it is necessary to implement in software. We provide concept development framework for application software programmable controllers, based on their integration with other systems (SCADA, SIS, BPCS, MES / MOM) and increased requirements for process diagnostics and maintenance.

Results and discussions. Developed a concept based on the implementation in PLC object model equipment(Figure 2), according to the concepts of ISA-88, ISA-95 and ISA-106. Each equipment entity is a functional block or function, and a set of data, that can realize the exchange with the upper level. The data structure and behavior of the function/FB compatible with defined in ISA-88, that is based on machines states, modes and interfaces defined in this standard. The procedural elements and basic control are also based on standard terms. That the developed framework is a library interrelated elements that ensure the implementation of the basic set of control module and equipment module, regardless of the control object and

determine the mechanism of their implementation in higher-level objects. The default equipment entity at level of control module are proposed to provide:
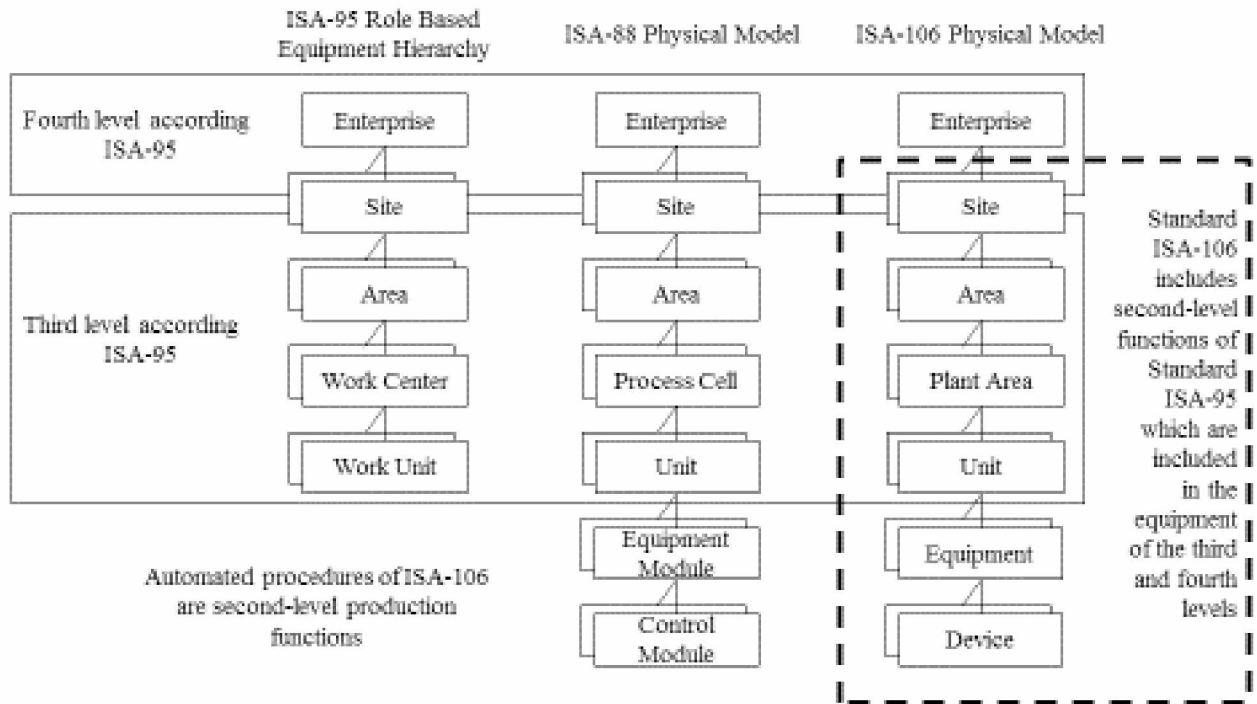

Figure 2. Physical Model

- controllers channel (DICH - discrete inputs, DOCH - discrete outputs, AICH - analog inputs, AOCH - analog outputs, COMCH - communication channels) for diagnostic channel and binding logical channels to physical;

- process variables (AIVAR - analog input, AOVAR - analog output, DIVAR - discrete input, DOVAR - discrete output) for complete information processing in process, including binding to channel, filtering, scaling, inversion, etc.; for easy debugging process; for functions of simulation modeling; for technological functions of alarm;

- Actuators (valves, control valves, motors, pumps): for control functions with feedback; for easy debugging process; for functions of simulation modeling; for technological functions of alarm; for conducting statistics.

All of the above elements in terms of the ISA-88 are the control modules, but in terms of ISA-106 are the devices. To unify the titles, we use the terminology ISA-88 as valid today. All control modules form a three-level hierarchy that allowed standard ISA-88(Figure 3).

The lowest level of control modules provides a level of abstraction from the hardware. That realization of this level depends on the chosen platform and method of implementation. Elements of the control module type "channels" are all arrays controller channels, regardless of their location and involvement in the process. Each array element is uniquely identified by a number, a binding to the physical channel is individually. The control module of the type "channel" their values are tied to the physical value of a particular channel and perform the following functions:

- Provide diagnostic information to higher levels of control modules such as a sign of authenticity, and the if possible reasons for refusal channel;

- Permit mode forcing value;

- Indicate the fact of binding to channel process variable.

The control module of the second level of the type "technological variables" can be attached to the channel of the same type by their number. Thus binding process variable to the channel is dynamic, allowing you to change the location of a particular physical connection of the sensor/actuators in the event of failure part of the system. In addition, this switch can be programmed. "Process variables" are higher in the hierarchy of controls then "channels". All diagnostic information are transferred from the "channel" to "process variables". Implementation of this level is independent of the hardware characteristics of the controller, since all platform-dependent subtleties implemented at "Channel" interface which is standard. "Process variables" provide the following functionality:

- Binding to the channel by its number;

- Disconnection of service (deactivation variable);

- Tracking authenticity of value for the error bound channel, out of range of measuring value, etc.;

- Diagnostics of the channel (transfer of diagnostic information from a connected "channel" on the upper level);

- Processing of input / output values: scaling (including piecewise linear interpolation), filtering, inverting;

- Availability of manual change mode (forcing) according to ISA-88;

- Availability of simulation mode in which to input variables value changes upper level control modules and for output variables, is freezing values output channels;

- Processing of alarms (ISA 18.2): Response thresholds for triggering consideration of delay, hysteresis, forming a common bit system alarm/warning;

- Configuring alarms processing (ISA 18.2): setting alarm values, types of alarms (alarm/warning/channel error), temporary removal from service alarm.
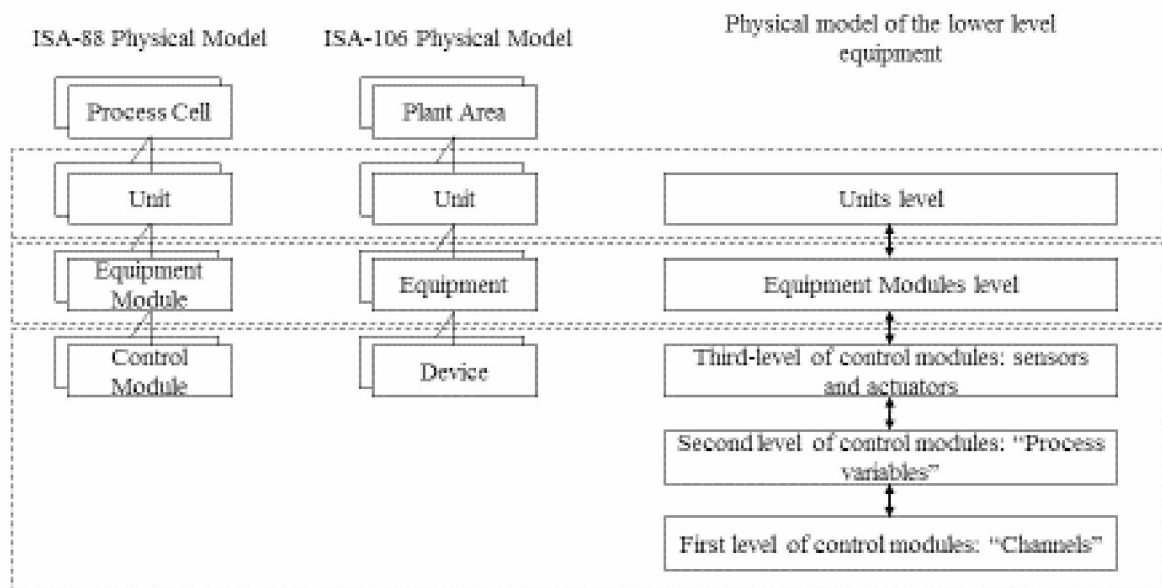

Figure 3. Physical model of the lower level equipment

The control modules of third level are the actuators, regulators, etc., and include basic control functions (according to the terminology of ISA-88). Each control module provides a two-way interaction with the "process variables" for recording and reading. This allows for a given level of implementation of specific functionality for a specific control module, and provide the following features:

- Take into account the state of "process variable" (normal/alarm/authenticity) and diagnostic information in the control logic execution control module;

- Simulation of the control module using it included modeling algorithm (if necessary):

o Advanced model-based diagnostic process;

o Model-based control;

o Work in simulation mode for demonstration/training or adjustment of system;

- Inclusion in simulation mode control module and all its related control modules lower;

- Statistical information (depending on the type of control module).

This approach requires a large amount of configuration (adjustment of objects) adjustment (steps, time step), diagnostic (reasons for rejection) that are not real-time data. Therefore, with the proposed concept is proposed mechanism of separation of real time data from the other, and exchange between SCADA/HMI and PLC via the buffer.

During the implementation of the proposed concept of having some difficulty and describes methods for their solution:

- The need for large amounts of memory and calculation $\rightarrow$ Modern PLCs have sufficient capacity;

- The need for high-network bandwidth PLC/SCADA $\rightarrow$ Total transition to Ethernet technology or separate traffic (group data in different memory areas PLC);

- A large number of tags, and thus more cost SCADA system → The configuration data is stored in the PLC, and if necessary sent to the SCADA system via a buffer variable.

According to this, all variables are split into real-time variables and configuration variables. PLC and SCADA system continuously exchange data real-time. The configuration variables are in memory PLC and sent in SCADA system only if necessary. This mechanism is implemented by the fact that each variable has two main attributes of status and command. When the system receives a command to the need to read the configuration variables required variable is loaded into the buffer and sent to the SCADA system. After the necessary manipulations were conducted with this variable, and the system receives a command to save, then the contents of the buffer is copied to this variable. This mechanism is shown in Figure 4.
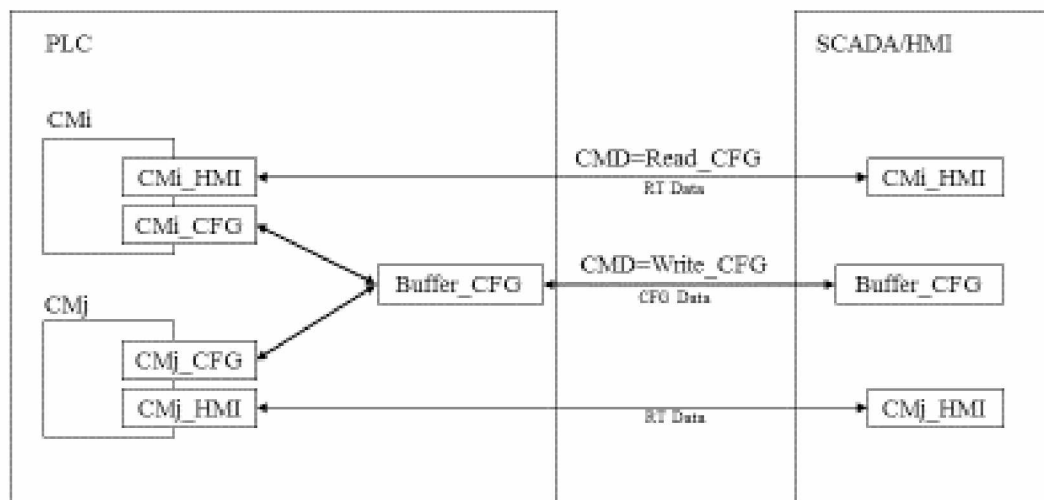

Figure 4. The mechanism of the buffer

Conclusion and prospects. This concept is now undergoing approbation of in several productions (platform S7 300 (Step 7 TIA Portal) + SCADA Zenon; platform S7 1200 (Step 7 TIA Portal) + WinCC Comfort) and in the laboratories of the department Integrated Automated Control Systems National University of Food

Technologies (platform Schneider Electric M340 (Unity PRO) + SCADA Vijeo Citect). Among the most significant advantages over older approaches to software development for PLC, there have been the following:

- Reducing the time of commissioning works;

- Reducing the time changes in project;

- Reducing the time malfunctions (some problems had not even manifested);

The above facts were shown only at a particular site and only at partial implementation of the proposed ideas. However, it should be noted that the development of the framework requires a lot of time, which offset a decrease in the time of its re-use and commissioning. The concept takes a lot of software resources, but to date modern PLC of PAC usually include much more than the resources used. Besides the savings from using the concept far exceeds the cost of capital. The functionality of the system is not limited to the stated above and may be extended.

References
1. Meyer, H., Fuchs, F., Thiel, K. (2009). Manufacturing Execution Systems. Optimal Design, Planning, and Deployment. The McGraw-Hill Companies, Inc., New York.
2. Kletti, J., Berres, B., Brauckmann, O. (2007). Manufacturing Execution Systems. MES, Springer Berlin Heidelberg, New York.

РОЗРОБКА ПРОГРАМНОГО КАРКАСУ ДЛЯ КОНТРОЛЕРІВ БАЗОВОЇ СИСТЕМИ КЕРУВАННЯ ПРОЦЕСОМ З УРАХУВАННЯМ ВИМОГ ДО ІНТЕГРУВАННЯ З ІНШИМИ ПІДСИСТЕМАМИ І РЕАЛІЗАЦІЇ СЕРВІСНИХ ФУНКЦІЙ ДІАГНОСТИКИ ТА ОБСЛУГОВУВАННЯ

О. Пупена, Р. Міркевич, О. Клименко

Анотація. *У роботі розглядаються підходи щодо розробки каркасу для прикладного програмного забезпечення програмованих логічних контролерів, як складових базової системи керування процесом (BPCS) з урахуванням їх інтеграції з іншими системами (SCADA, SIS, BPCS, MES/MOM) і підвищених вимог до діагностики процесу та сервісного обслуговування.*

Концепція програмного каркасу розрахована на контролери рівня PAC (Process automation controller), базуються на відкритих стандартах IEC 61131 і не використовують специфічні для виробниках рішення. Структура та внутрішня реалізація бібліотечних елементів каркасу базуються на ідеях ISA-88, ISA-95, ISA-106 (Draft) яка передбачає максимально просту інтеграцію системи керування в єдину інтегровану автоматизовану систему керування.

У роботі запропоновані механізми покращення діагностування процесу та самої системи, швидкого налагодження системи та відновлення її функціональності після збоїв. Каркас є взаємопов'язаним набором бібліотечних елементів функцій та функціональних блоків, які описані на рівні їх інтерфейсу, принципів взаємодії та алгоритмів функціонування. Передбачається, що для реалізації каркасу для конкретного контролеру будуть використовуватися мови IEC 61131-3, але це не є обов'язковою умовою. Набір елементів каркасу можуть розширюватися і добавлятися додатковим функціоналом без порушення загальних принципів.

Вперше запропоновано універсальну структуру управління процесами молочного виробництва з використанням сучасних стандартів побудови інтегрованих автоматизованих систем управління.

Ключові слова: ***BPCS, ISA-88, ISA-95, SIS, каркас***

РАЗРАБОТКА ПРОГРАММНОГО КАРКАСА ДЛЯ КОНТРОЛЛЕРОВ БАЗОВОЙ СИСТЕМЫ УПРАВЛЕНИЯ ПРОЦЕССОМ С УЧЕТОМ ТРЕБОВАНИЙ К ИНТЕГРИРОВАНИЮ С ДРУГИМИ ПОДСИСТЕМАМИ И РЕАЛИЗАЦИИ СЕРВИСНЫХ ФУНКЦИЙ ДИАГНОСТИКИ И ОБСЛУЖИВАНИЯ

А. Пупена, Р. Миркевич, О. Клименко

Аннотация. *В работе рассматриваются подходы к разработке каркаса для прикладного программного обеспечения программируемых логических контроллеров, как составляющих базовой системы управления процессом (BPCS) с учетом их интеграции с другими системами (SCADA, SIS, BPCS, MES/MOM) и повышенных требований к диагностике процесса и сервисного обслуживания.*

Концепция программного каркаса рассчитана на контроллеры уровня PAC (Process automation controller), базируются на открытых стандартах IEC 61131 и не используют специфические для производителях решения. Структура и внутренняя реализация библиотечных элементов каркаса базируются на идеях ISA-88, ISA-95, ISA-106 (Draft), которая предусматривает максимально простую интеграцию системы управления в единую интегрированную автоматизированную систему управления.

В работе предложены механизмы улучшения диагностики процесса и самой системы, быстрого налаживания системы и восстановление ее функциональности после сбоев. Каркас представляет собой взаимосвязанный набор библиотечных элементов функций и функциональных блоков, которые описаны на уровне их интерфейса, принципов взаимодействия и алгоритмов функционирования. Предполагается, что для реализации каркаса для конкретного контроллера будут использоваться языки IEC 61131-3, но это не является обязательным условием. Набор элементов каркаса могут расширяться и добавляться дополнительным функционалом без нарушения общих принципов.

Впервые предложено универсальную структуру управления процессами молочного производства с использованием современных стандартов построения интегрированных автоматизированных систем управления.

Ключевые слова: ***BPCS, ISA-88, ISA-95, SIS, каркас***