

Jacek Augustyn\*, Maciej Klemiato\*, Jan T. Duda\*\*

## **Reconfigurable Integrated Production Control System of Industrial Processes on QNX Neutrino Platform\*\*\***

### **1. Introduction**

Modern industrial systems impose high requirements on automatic control quality. These requirements are often impossible to fulfill with the use of classic methods, that is why computer control systems are readily used to the wide range of control applications. Continuous increase in computing power enables using of more and more sophisticated algorithms of control and optimization. Advanced methods, considered only theoretically until recently because of too long computing time, have now practical applications. In that way the quality of control is highly improved and, following on from this, efficiency, costs and security of production are improved too.

Computer control systems consist of the three main levels:

- 1) data acquisition and direct control,
- 2) supervisory control,
- 3) management.

The aim of the lowest level is providing a process with the uninterrupted control using simple but reliable algorithms. This level makes direct measurements of the process variables and computes the set-points for actuators.

The aim of the supervisory control is coordination and supervision of the direct control. These tasks are mainly conducted by operators and a computer system delivers only information needed for taking actual decisions. Therefore, process visualization and data acquisition plays here the main role. Systems playing only such role, called SCADA (*Supervisory Control and Data Acquisition*), are now the majority of installed computer control systems of the supervisory level [1].

---

\* AGH University of Science and Technology, Faculty of Electrical Engineering, Automatics, Krakow, Poland

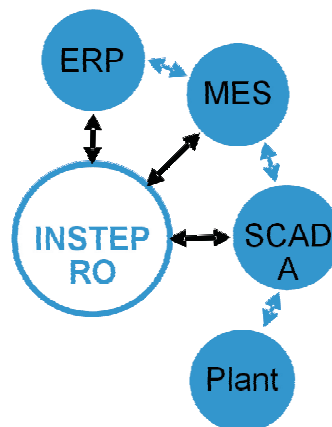
\*\* AGH University of Science and Technology, Faculty of Management, Krakow, Poland

\*\*\* The paper was prepared within the grant UDA-POIG.01.03.01-12-171/08/00 founded by the European Regional Development Fund in Innovative Economy Programme (Priority Axis 1.3.1)

Management level plans production based on consumer's orders and available production resources (machines, people, stores, raw material, intermediate and final products) and supervises usage of these reserves. The complexity of decision problems at this level and the impossibility of taking into account all the potential cases, causes that control has, as a rule, advisory character. The decisions made need verification by an operator of the process.

The presence of the operator and direct control systems in the majority of manufacturing companies have become a standard in recent years. Direct control systems usually use PLC (*Programmable Logic Controller*) and PAC (*Programmable Automation Controller*) equipped with analog and discrete input-output modules. Supervision and visualization is accomplished by the use of PC computers and SCADA software. At the higher level of management, MES (*Manufacturing Execution Systems*) are used for production scheduling and short-distance planning. The present MES connect two aspects of an enterprise – production, which runs under the time regime of seconds and business, based on long-, medium- and short-term strategies, planning, finance control, etc., which are longer periods (days, months, quarters) [5].

At the highest decision level many manufacturing companies have ERP (*Enterprise Resource Planning*) or MRP (*Material Resource Planning*) systems. They contain procedures for managing the whole enterprise (financial investments, marketing, human resources, orders for production and resources, inventory monitoring, sales, transport, etc.) and are intended for management teams. MES are the intermediate layer between business levels (such as ERP) and direct control level, devices and technical apparatus. Besides tasks such as ASP (*Advanced Planning and Scheduling*), MES also deal with recipe management, production reporting, crisis management, material resources, inventory tracking and deliver KPI (*Key Performance Indicators*). As far as ERP is mainly applicable at an office (executive staff), MES's recipient is usually a shop floor (technical staff).



**Fig. 1.** The INSTEPRO system against the background of control/management systems

Despite the availability of many decision and production support systems, enterprises indicate that there is a need for software which fill the gap between business management systems and the real situation at a shop floor. In response to that signal, the AGH Automatics Department brought to life the idea of the INSTEPRO project – *Integrated Production Control System*. The aim of this project is to make a software tool for the monitoring, control and supervision of production processes in manufacturing companies which are already equipped with PLC/SCADA and ERP systems (Fig. 1). The INSTEPRO system, which is the subject of this article, will provide tools for automation of production processes management with the use of advanced algorithms worked out at the AGH Automatics Department.

## 2. Requirements of the INSTEPRO system

The INSTEPRO software allows to simultaneous control (data acquisition, computing, sending control variables to the direct control level), data archiving (process variables, control variables and other relevant data for further analysis) and visualization of process variables on an operator's displays. Running of the crucial tasks (data acquisition, control, optimization) shouldn't be interrupted by other system functions, especially by visualization tasks. However, the system should provide suitable comfort for operators by fast access for all the necessary data and systematic reports actualization. It is obvious that, because of the complexity of the algorithms accomplishing certain functions (control, data acquisition, operator interfaces), implementating the system by just one program would be difficult or even impossible. While it seems reasonable that separate processes or groups of processes, which communicate each other and use a common database, should accomplish individual tasks. If these processes are running on one computer we talk about concurrent computing (with time scheduling). Modern systems more and more often apply parallel processing (distributed), in which processes are computed on different network nodes. It lets, among others, fulfill contradictory requirements mentioned above (regarding essential and operator tasks) by computing crucial functions on workstations whereas visualization functions on a separate console terminals. To enhance reliability, each of these stations should have the complete software installed, enabling to run all the system functions. Increase in computing power of the system by using such approach depends on computing time to data exchange (between processes) time ratio [2, 3].

Taking into account the above, the INSTEPRO system was equipped with specialized modules such as:

- process data exploration (including time series analysis for early warning of unusual situations and malfunctions, event studies in process variables, diagnostic analysis),
- process optimization,
- simulation of nonlinear dynamic processes,
- identification of static and dynamic models,
- advanced control algorithms (including multidimensional predictive control).

Additionally, to ensure that INSTEPRO meets the requirements of such systems and is useful for the industrial process, it should provide the following:

- scalability, defined as adding extension modules (e.g. strictly dedicated for an industrial process) and adding computing power,
- integration with other systems, especially with PLC/SCADA layer and MES, MRP as well,
- scenario simulation of different control strategies,
- dedicated real-time databases (of process variables) handling,
- advanced control algorithms implementation,
- archiving a big amount of process data, which can be then analyzed for technology and control algorithms enhancing,
- implementation of historical data processing algorithms and the logical analysis of process variables for diagnostics and expert systems,
- adapting the system to the needs of different industrial processes (versatility),
- simulation of different parts of the system without the need of suspending the production process,
- working in computer network, which allows to control very large installations and distant objects,
- implementation of supervisory control for existing direct control systems,
- adapting the system to changes in the production profile, devices replacement, the needs of the market, etc. (flexibility),
- connecting supervisory control subsystems with production management systems.

Detailed assumptions and formal principles regarding crucial computing tasks of the INSTEPRO system was presented in [4].

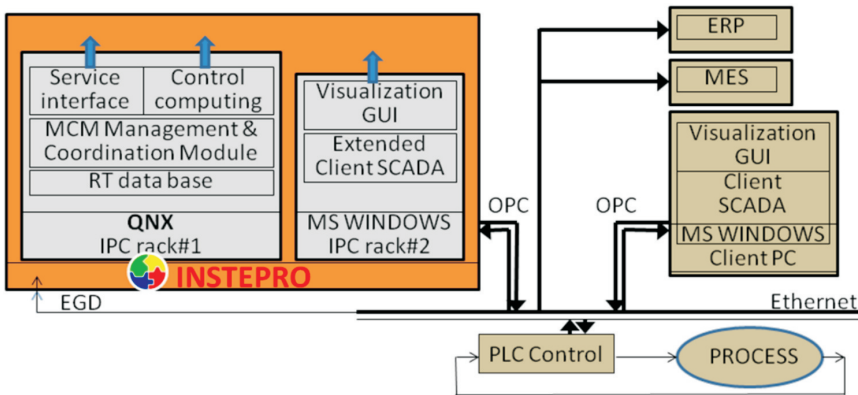
### **3. Realization of the solution**

Regarding the assumptions and nature of the system, its main feature is a possibility of control algorithms and data processing reconfiguration. The system is modular, hence its extension of new functions is relatively simple. The software allows for flexible adjusting to variety hardware configurations of direct control. Particularly, there is a possibility of easy adaptation for slow-changing continuous processes control (e.g. distillation, heating, glass production).

#### **3.1. Architecture of the solution**

The INSTEPRO system works on two-board two-system industrial PC computer [8]. The operating system is QNX Neutrino because many tasks of the INSTEPRO run in real-time regime (considering assumed control cycle). QNX is a modern real-time operating system, allowing of scaling solutions and distribute computing on many computers in a network [6, 10]. It also provides task isolation – tasks run in separate address spaces, which increase stability. Failure of one module doesn't destabilize the whole system.

The overall architecture of the environment is presented in Figure 2. On one board of the IPC computer, depicted as IPC rack#1, QNX Neutrino operating system is installed. It includes the main INSTEPRO system. The second board (IPC rack#2) plays supporting role. It is equipped with the MS Windows operating system for running a copy of the end-user SCADA system. It provides visualization of the supported industrial installation. SCADA systems have reach graphical capabilities and libraries, therefore making additional graphical user interfaces in QNX seems to be ineffective. The SCADA application is slightly modified in order to make it easier for an operator to adjust essential parameters of the INSTEPRO system.



**Fig. 2.** The overall architecture of the solution. Arrows mean data flow

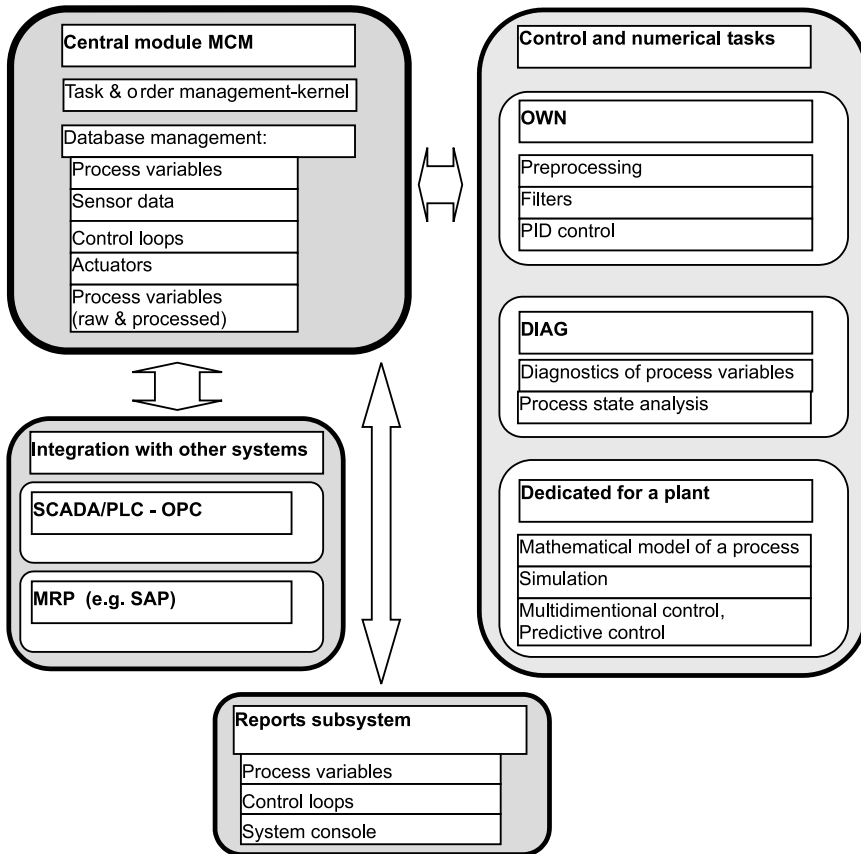
Such architecture reflects the way of application design called *shadow computing*. Its main assumption is non-invasive integration with an existing control system. The INSTEPRO system can work at the background using existed process data. At the beginning it can work in the advisory mode, when the computed control variables or recommendation of changing the process parameters are only presented on a modified SCADA window. An operator can approve this recommendation manually and apply them in the existing control system. However, the automatic mode is also possible. In this mode the operator has to accept the computed control variables only once, allowing them to be applied in the control system.

### 3.2. The detailed architecture

To meet the requirements defined in the chapter 2, it was assumed that the INSTEPRO system will consist of many independent but cooperating modules. They are grouped in four main functional areas:

- process management and coordination (MCM, management and coordination module),
- crucial computing tasks (identification, optimization and control),
- integration with SCADA and ERP with the use of OPC protocol [9] and dedicated communication module in EGD standard (*Ethernet Global Data*) [7],
- reporting subsystem.

The architecture of the system tasks was depicted in Figure 3. The MCM module is responsible for starting all others modules. It can restart or reinitialize them if necessary. It also provides time synchronization of all others tasks and maintains task priorities. Additionally, it delivers databases with configuration parameters for running, sensor data, process variables and control loops. The database of the process variables plays a superior role, it contains raw and processed time series.



**Fig. 3.** The model of the functional groups and the tasks of the INSTEPRO system.  
The arrows mean data flow

The INSTEPRO software is an integrated computing-decision system. The tasks can be divided into two main types, depending on their functionality. First type processes are directly connected with the operating system and provide task coordination, data exchange between tasks and management of the whole system. The first type processes are the main module MCM, which manages the operating system functions, and dedicated libraries, which are intended to isolate a user from the operating system and inner communication

protocols. The second type processes are automatic specific. These subsystems accomplish essential functions such as: data processing, control, monitoring, identification, process state analysis, alarms managing, data archiving and the handling of operator commands.

The main module (MCM) can check if a particular task exists and what is its state. To enhance the efficiency of the whole system, it is possible to create or kill tasks dynamically. It saves the operating memory and minimizing the effects of various errors (e.g. premature ending or blocking of a running task). The task management is very important in distributed control – after device damage or shutdown, tasks are taken over by a good working node. There is a precisely defined list of tasks which can be accomplished in the control system. Each task has exactly specified functions and resources which can be demanded and ways of communication with other tasks. Such a structure of the system provides its reliability and let avoid crashes.

### 4. Module structure

As there are many tasks which carry out important numerical algorithms, the standard rules of cooperation between them was established. The methodology, explained below, was adopted to facilitate programming work.

Essential functions are called *system orders*. Each order corresponds to the procedure, which accomplish or supervise particular operations. This is so called *order procedure*. It is usually completed in a relatively short time, without the inner supervision of the time regime (e.g. one-time computing control inputs by a control algorithm, one-time report update, solving of an optimization task, one-time alarms analysis, etc.). These procedures are grouped into tasks. It is possible to add a new order (enhancing the system functionality in this way) by adding to the source code a new order procedure (possibly with auxiliary procedures) and a code which calls this procedure (see Fig. 4).

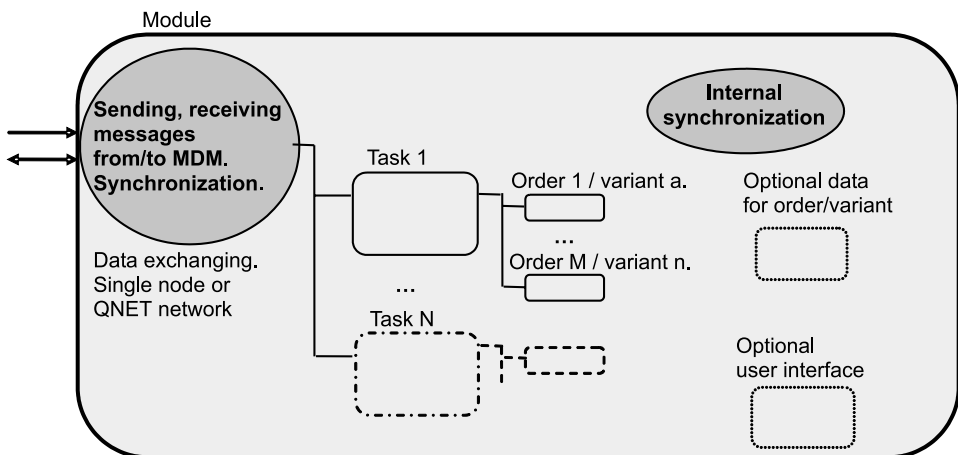


Fig. 4. The single module architecture

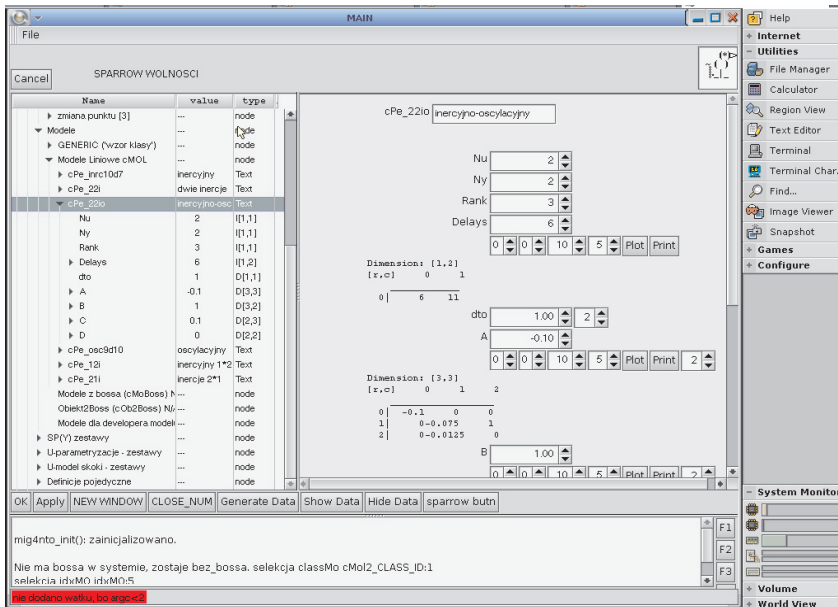
The assumption was made that tasks often activated reside in the computer operating memory, whereas these which are used seldom are read from a hard drive to the memory and after completing removed. Orders use procedure libraries dedicated to communication between tasks and facilitated programming complex functions. Tasks executing various orders relate only by the shared database (system modularity), however, orders contained in the same task can use common local data structures.

The computing modules and the main module can work at different network node. Using a computer network with the use of QNET [11] and the functioning of the software in the distributed mode can be very crucial in the control of big and even moderate industrial processes.

There are the following operating modes of the module:

- 1) *Slave* – cooperates with the main module of the INSTEPRO and shares data with it. Slave is fully synchronized and managed by the main module.
- 2) *Master* – cooperates with the main module of the INSTEPRO and shares data with it, but doesn't depend on its time synchronization nor management.
- 3) *Stand-alone* – the module works totally independently of the main module.

The first mode is a preferred operating mode. The other two modes are used in the development stage. The stand-alone mode allows to integrate with others managing systems as well.



**Fig. 5.** The interface for predictive control and model parameters configuration, made in the Photon QNX Neutrino environment



Each of the modules can work as a numerical computing server for the main module in, so called, *silent mode*. Most of the numerical modules don't have the user interface. It enhances reliability and can reduce delays during the functioning. Tasks which demand many configuration parameters, and during development, can optionally have the user interface. Exemplary interface for the predictive control and model parameters configuration is presented in Figure 5. After starting the system, the interface can be activated or deactivated on demand.

## 5. Conclusions

This paper presents assumptions and the key elements of the reconfigurable integrated production control system. It is made in the real-time environment – QNX Neutrino. The discussed system is dedicated to chemical, metallurgical, heating industry and other continuous industrial production processes.

The work is being currently implemented in the glass melting plant control system. The implementation includes system configuration and adaptation to existing process variables and technological parameters. Process variable acquisition from the existing control system is made by the use of the OPC protocol.

The main control object is the glass feeder. The most important control problem is the stabilization of the three-dimensional temperature gradient of the liquid glass at the end of the glass feeder. The feeder is a 12 meters long, multidimensional object with 30 process parameters measured (mainly temperatures). Actuators consist of heating and cooling systems of subsequent feeder sections, both for the right and the left side. The aim of the supervision control, fulfilled by the INSTEPRO system, is optimization of the operating points changes, during work parameters modifications (e.g. in production assortment changing). The main objective is to minimize the time of the transient states.

Additionally, integration with the existed ERP system in the company will allow to consider production plans from the management level and reporting of the control possibilities in return.

## References

- [1] Bailey D., Wright E., *Practical SCADA for industry*. Burlington, Newnes, 2003.
- [2] Byrski W., *Obserwacja i sterowanie w systemach dynamicznych*. Uczelniane Wydawnictwa Naukowo-Dydaktyczne AGH, Kraków, 2007.
- [3] Duda J.T., *Modele matematyczne, struktury i algorytmy nadrzednego sterowania komputerowego*. Uczelniane Wydawnictwa Naukowo-Dydaktyczne AGH, Kraków, 2003.
- [4] Klemiato M., Augustyn J., Duda J.T., Sterna K., *INTEPRO: Zintegrowany System Sterowania Produkcją (2)*. PAR, 5/2011.
- [5] Meyer H., Fuchs F., Thiel, K., *Manufacturing execution systems: optimal design, planning, and deployment*. McGraw Hill Professional, 2009.

- [6] Ułaszewicz J., *Systemy czasu rzeczywistego QNX6 Neutrino*. Wydawnictwo BTC, Warszawa, 2007.
- [7] *Ethernet Global Data Objectives*. GE Fanuc Automation Americas Training Services [<http://geplc.com/downloads/Labs/GFS-384%20M09%20EGD.pdf>].
- [8] *MS-9643 Mainboard*. User Manual, rev. 1.3. Micro-Star International, 2007.
- [9] OPC Foundation [[www.opcfoundation.org](http://www.opcfoundation.org)].
- [10] *QNX Neutrino RTOS. System Architectures*. QNX Software Systems GmbH & Co, Ontario, 2009.
- [11] *QNX Neutrino Realtime Operating System. Programmer's Guide*. QNX Software Systems GmbH & Co, Ontario, 2009.