

## INTEGRATED ENVIRONMENT FOR ACQUISITION, PROCESSING AND CONTROL BY DIGITAL DATA

© 1995, S. Kiryukhin, I. Kutsevich, N. Kutsevich, M. Pertsovsky, Ye. Vorobjov

*Institute of Radioengineering and Electronics of the Russian Academy of Sciences (IRE RAS)*

Total solutions for PC-based industrial and lab-automatization include:

- Development of software and hardware components for real-time systems;
- Development of handlers for non-standard external equipment;
- Development of algorithms for generation of program code for supporting interaction with external devices;
- Development of graphical environments under Windows using object - oriented technology;
- Development of turnkey systems.

For implementation of these problems IRE RAS developed the following basic components:

1. Real-time plug-in boards for:
  - digit input/output to increase throughput of data acquisitions systems;
  - parallel digital input to test complex devices;
  - nonmasked interrupt controller to reduce a response time for external events.
2. Interface equipment libraries for both standard (CAMAC, GPIB etc.) and non standard (real-time plug-in) boards to support interaction with external devices.
3. Instruments for generating program code to control external devices.
4. Integrated environment (IE) of problem-oriented automated systems, presenting the instruments to use above mentioned possibilities.

### Integrated environment (IE) structure

The structure of the IE is described in [1] and represented on the fig.1.

The IE consists of the following modules:

- flowchart's graphic editor. It includes the means for drawing flowcharts to define program's algorithm. The flowchart is combined from objects - an external devices for PC or program modules;
- module for interaction with IE's data base (DB). DB includes the formal descriptions both for standard interfaces controllers and for real-time boards, using frame representation and production rules. The module allows to represent board's formal descriptions from DB and modify them;
- generator of program fragments. It realizes the mechanism of the DB information analysis to receive program code on the traditional language;
- debugger and graphic representation modules.

These tools allow to run some operation without generating program code and to represent the results using textual and graphic form.

Each of mentioned components is independent product and can be used for software development separately. But the components in the environment are complementary and comfortable to develop the automatization system.

Unity of multifunctional components in IE is supported by using object-oriented technology (OOT). In our opinion the most complex and interesting problem is the development of IE's real-time component. Therefore in this article using OOT is considered for real-time realization in IE for example. This part of IE is pointed on fig. 1 by dash lines.

We keep in mind that task is a part of program, doing some logical indivisible action and data, used by program; a task is unit for dispatching and uses system's resource. Proposed dispatcher allows to compensate MS DOS disadvantages partly.

### Implementation technology

The realization of traditional mechanism for switching tasks in the program's development environments for

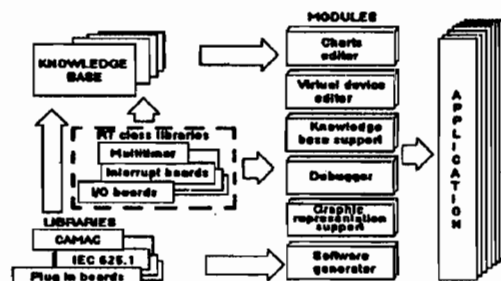


Fig.1 Structure of integrated enviroment for experiment automation systems

PC is possible using specialized interval timer (analog – chip 8253 Intel). Timer intervals precision is 0.5 mks.

Possibility of simultaneous trace for several timer intervals is defined by the amount of chip's free channels. In the IBM PC the dispatcher can use only channel of system timer. That dispatcher realization leads to reducing precision on the task's switch. One of the ways is realization of multitask dispatcher on the base of special multitimer. It can trace a lot of timer intervals simultaneously (up to 256) and the output signals. They can be used as interrupts sources or control signals with high precision (0.3 mks).

Dispatcher's main characteristics:

- installation and start of the tasks;
- stop and resume of the task;
- abort of task;
- control by priorities: writing and reading current priority;
- control by the task: stopping current task on some time or waiting some moment of time;
- synchronization of exchange using the mechanism of semaphores;
- asynchronous exchange of data between the tasks.

The dispatcher's class inherits a properties of timer's and multitimer's classes, supporting different mechanisms for simultaneous trace of several timer signals (fig.2). For multitimer's class the first three groups of characteristics can be given at absolute of time relative (with initialization moment of multitimer or regarding some external event).

One of the base classes is the class TASK. It contains the status information about the task and the functions for its modification. The functions of the class TASK SCHEDULE allow to view any status information for each task from the current application, to modify the task's priority, to activate and stop some task, to determine the application's structure and etc.

A set of functions dispatcher's class includes next groups:

- functions supporting the interaction mechanism with multitimer, accessible for dispatching only;
- functions, defining application's configuration. They

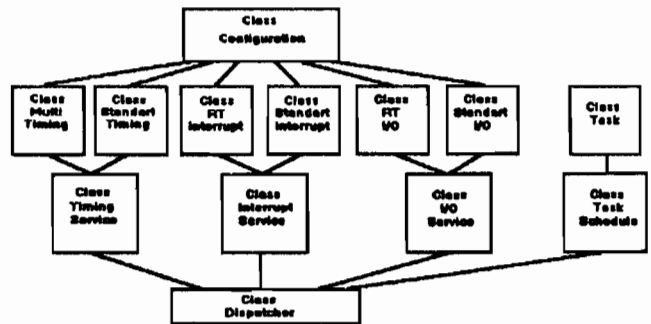


Fig.2 Hierarchy of classes for multitask application

- functions for dispatching tasks. The first group of functions and this group guarantee the minimum for tasks switch;
- functions to control by tasks. That group is the most important for programmers, it supports the mechanism of tracing tasks.

### Conclusion

Through usage of OOT for the development of all IE's components allows to fast adapt the total solutions to user's requirements.

Using hierarchy of classes allows to build the multitask applications; their structure has hardware and software, taking into account the RT requirements.

Using the IE allows the user to participate directly in the real-time systems development and their installation. Software is developed under Windows.

### References

1. S.Kiryukhin, N.Kutsevich, M.Pertsovsky Hardware and software support for Laboratory experiment automation Systems using IBM PC/AT // Proceedings of Vera+ & VITA CONFERENCE VVConex 93-Moscow, 1993.-P.200-204.

## ИНТЕГРИРОВАННАЯ СРЕДА ДЛЯ СБОРА, ОБРАБОТКИ И УПРАВЛЕНИЯ С ПОМОЩЬЮ ЦИФРОВЫХ ДАННЫХ

С. Кирюхин, И. Кучевич, Н. Кучевич, М. Перцовский, Е. Воробьев

Общие решения для автоматизации на базе персональных компьютеров включают:

- разработку компонентов программного и аппаратного обеспечения для систем реального времени;
- разработку устройств обслуживания нестандартного внешнего оборудования;
- разработку алгоритмов формирования программного кода для поддержки взаимодействия с внешними устройствами;
- разработку графической Среды под управлением операционной системы Windows с использованием технологии, ориентированной на пользователя.