A Multi-threaded LabVIEW™ Data Acquisition and Control Program for a Laboratory Reactor System

by
David Thomson
Systems Integrator
Original Code Consulting
and
Martin Guinn
Principal Development Engineer
Roche Colorado Corporation

Category:
R&D

Products Used:
LabVIEW
Automation Symbols Toolkit
FieldPoint™

The Challenge: To develop a computer interface for a laboratory chemical reactor system which would integrate numerous instruments, sensors, and valves and would present the operator with an intuitive graphical representation of the physical system.

The Solution: The LabVIEW programming environment was used to create an easy-to-use interface to the laboratory reactor system. FieldPoint distributed I/O products were easily integrated with LabVIEW to supplement the existing instruments. The inherent parallelism and the multithreading capabilities of LabVIEW were utilized to achieve the high performance required.

Abstract :
Roche Colorado Corporation required a computer interface for control and data acquisition of a laboratory chemical reactor system. The system consisted of two parallel reactor vessels, each with a stirrer and a temperature-controlled circulator. In addition, the two reactors shared use of a vacuum pump controller, a dosing pump, a balance, and a temperature-controlled condenser circulator. A LabVIEW program was written which integrated these instruments as well as 10 additional sensors and eight valves. This program included a configuration manager, a simulation mode for testing and demonstration, and a multi-threaded data server architecture for high-performance and high-reliability operation.

Introduction
At Roche Colorado Corporation, engineering process development includes scale-up studies using a laboratory mini-plant designed to resemble production equipment in geometry and operation. This system consists of two glass, jacketed reaction vessels (2-8 L), each of which can be temperature controlled from –20 °C to 150 °C by a Julabo circulator and stirred by an IKA stirrer (Figure 1). Additional equipment is shared by the two systems, including a Vacuubrand vacuum pump controller for pressure control, another Julabo circulator for controlling the coolant temperature of a condenser vessel, a KNF dosing pump, and a Mettler balance for metered input of reactants. Until recently, this system was operated manually. Each instrument was controlled through its own front panel. Although some instruments could be programmed with limited recipes, the operator had to coordinate the system, and data was recorded manually.

Figure 1. Photograph of the two-reactor system at Roche Colorado Corporation. The 8 liter reactor is at the lower left, and the 5 liter reactor is near the center. Each has a blue IKA stirrer controller above it. The Vacuubrand vacuum controller is on top of the two blue boxes between the stirrers. One Julabo circulator can be seen at the lower left: the others are off the right side of the picture.

It was recognized early that reliability and productivity would be increased if the reactor system were brought under computer-based data acquisition and control. Previous experience with commercial software packages available with turnkey laboratory reactor systems, such as Mettler's LabMax, revealed several undesirable limitations. These programs were generally expensive, difficult to alter, and used inflexible recipe-driven control architecture. In an attempt to overcome these shortcomings, it was decided to use LabVIEW to write a custom data acquisition and control program for the reactor system.

Laboratory Reactor Data System

One of the advantages of using LabVIEW is the ease with which a graphical user interface can be developed. For the Laboratory Reactor Data System, the Automation Symbols Toolkit was used to develop a front panel that presented the operator with a graphical representation of the physical system (Figure 2). The reactor vessel, circulators, and pumps were connected with pipe indicators to display the flow of fluids through the system. The pumps, stirrer, circulators, and valves were indicators that change colors to indicate their on/off state, and the pipes are colored according to which valves are open and what fluids are flowing through them. In addition, invisible controls were overlaid on the pumps, circulators, stirrer, and valves so that clicking on one of these instruments would toggle the on/off state of that device.

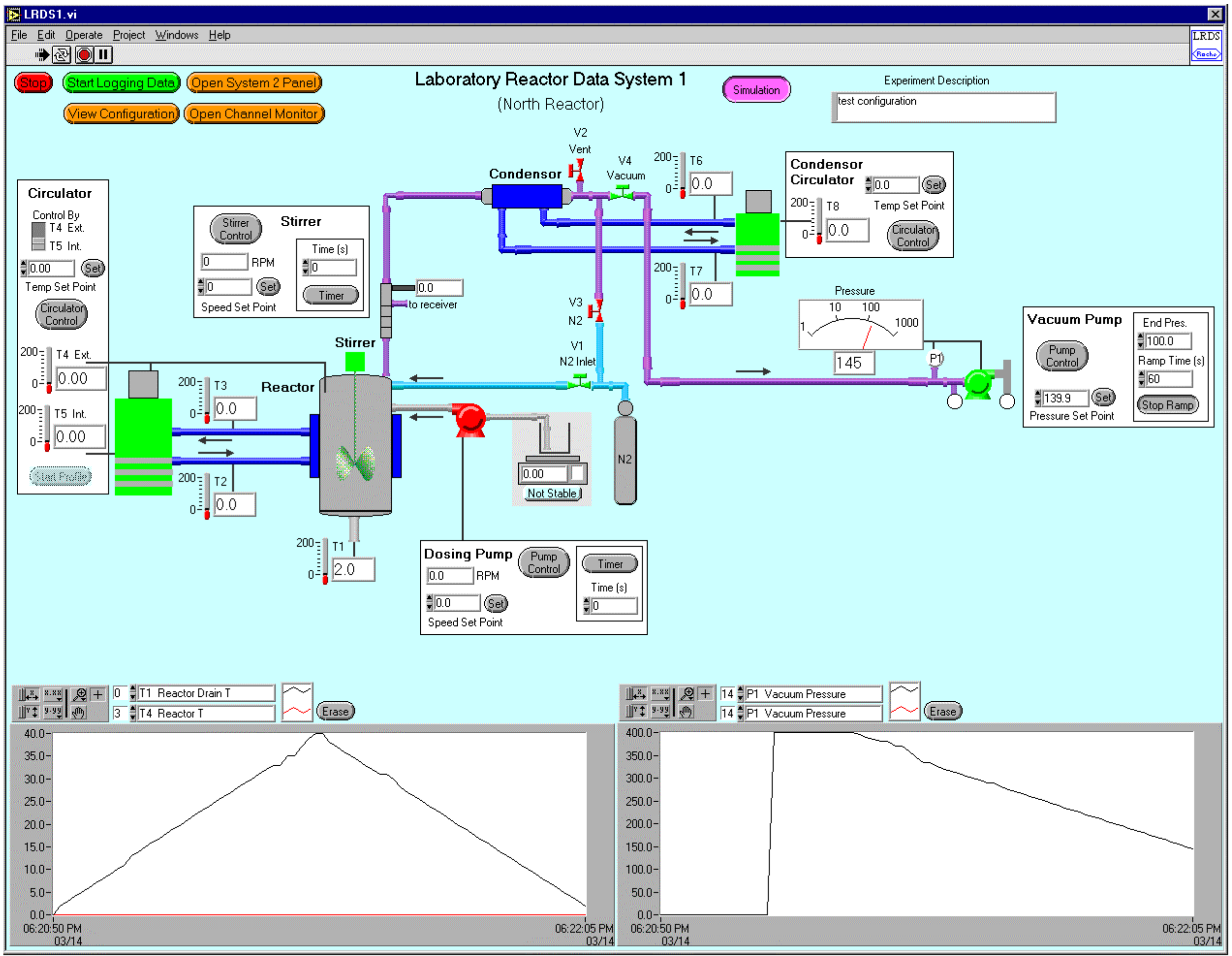

Figure 2. Front panel of the Laboratory Reactor Data System.

Important system parameters were also given indicators on the main front panel. Temperatures measured by the Field Point modules were displayed with thermometer indicators, and the vacuum pressure and balance readings were given indicators resembling those on the physical instruments. Control over the most often used parameters (set-points and timers) was also supplied on the front panel. Two charts at the bottom of the panel allowed any four data channels to be graphed for the duration of an experiment.

Multithreaded Data Server Architecture

The types of processes controlled by the Laboratory Reactor Data System typically have time constants that are long enough that a one second or longer cycle time is quite adequate. Even at this relatively slow rate, however, performance can become a significant issue when dozens of data values must be acquired during each cycle from relatively slow instruments with serial interfaces. To ensure adequate performance, a data server architecture was used. When the main front panel is run, it opens a second top-level VI in the background. This data server VI runs independently in parallel to the main VI. The data server and its sub-VIs run in separate threads from the main VI and at a higher priority. This ensures that the data server acquires and stores data on a regular interval, regardless of how the operator might be interacting with the main VI. In particular, if the operator has a file dialog or other dialog open, the main VI will be waiting for that dialog to close, but the data server will continue acquiring data and writing it to disk. The data server VI also writes the data to a global variable, which the main VI accesses for displaying the current process variable values.

The data server VI uses additional parallelism to maximize performance. Requests for data from the FieldPoint modules and from the eight different instruments are all made in parallel. Since some of the serial instruments can take 10s or even 100s of milliseconds to respond to a request for data, and since up to four data values are obtained from each instrument, this parallelism is required to attain a one second cycle time. Furthermore, the multithreading parameters of the LabVIEW environment must be adjusted to allow eight parallel serial port calls to work efficiently. The default thread configuration in LabVIEW is one thread for each priority of each execution system. For the Laboratory Reactor Data System, the threadconfig.vi was used to enable eight threads in the above-normal priority of the Instrument I/O execution system. This configuration allows the operating system to efficiently share CPU time between each of the eight parallel serial port calls while they each wait for data to appear in response to their commands.

Conclusions

The Laboratory Reactor Data System uses a LabVIEW program to control a dual chemical reactor system that includes eight instruments and numerous Field Point channels. By using a multi-threaded data server architecture with a high degree of parallelism, reliable operation and high performance are obtained. The new automated reactor system allows reactions to be controlled more precisely and studied more carefully. Furthermore, the use of the LabVIEW development environment produced an economical solution that is expandable and easily maintainable in the future.