BIOREACTOR MEASUREMENT AND SIMULATION ENVIRONMENT

K. Salonen, K. Kiviharju and T. Eerikäinen

Helsinki University of Technology
Laboratory of Bioprocess Engineering

Abstract: The aim of the work was to generate a flexible measurement, control and simulation environment for laboratory and pilot scale experiments. Various measurement devices were connected and synchronized to an integrated system with many bioreactors. Integrated devices were for example biomass monitoring, mass spectrometer and glucose control. Microcontroller-based interface modules and device drivers capable of transferring data to Multi Fermentor Control System (MFCS) via software interface were developed for each integrated device. A simple network TCP/IP interface enabled to distribute measurement devices in the laboratory intranet. The final result was an enhanced and flexible bioreactor data acquisition system.

Keywords: Fault diagnosis and monitoring, Sensors and Soft Sensors, Dynamics and control.

1. INTRODUCTION

The bioreactor cultivation is the most critical unit process, which can be modelled, simulated and controlled with various approaches like statistical experimental design, kinetic modelling and metabolic modelling. Appropriate measurements and data collection are essential for proper modelling, and a large amount of effort may be needed to make proper measurement, monitoring and sampling during bioreactor cultivations and down stream processes.

Significant progress of computation science has made it possible to collect and process data with much less effort than earlier. Automation has spread to all areas of science, even to basic laboratory research. Programmable logics and microcontrollers have brought the computer science very close to the physical applications. The computer world is integrating into networks and sharing of information is effortless (Bertocco et al., 1998). These relatively new technologies are easy to apply also to the area of biotechnology and especially bioreactor cultivations. Still it seems, that putting all measurement together from different measurement sources and distributing the data with various users is not trivial neither cheap due to different connection and formatting standards.

2. SYSTEM DESCRIPTION

Data integration is an essential requirement for online process control in bioreactor cultivations. It is also a way to decrease handwork required in collecting and processing data before the final man made data analysis. In general it isn't obvious that all data are collected online at one place and yet shared online with other applications, such as Matlab/Simulink (MathWorks) based distributed controllers. Almost all laboratory devices used in monitoring and control of bioreactors have some kind of a physical interface for external communication. To be able to communicate with PC the measurement signal has to be in a digital form. Some devices, however, have only an analogue interface. In these cases additional hardware is needed. Most interfaces differ from each other so that a unique solution is required for each device integrated to the system. Flexibility is an important element in a multi bioreactor system. It has to be possible to use portable special devices (e.g. analyzers) with any bioreactor at any time without difficulties. Another necessary demand is modularity, which enables modifying and expanding the system with minimum effort. These demands led to practical problems: How to integrate devices with highly different interfaces to the system so that the system would stay as flexible as possible? A goal was to find a general method that could be applied with small modifications for all devices. This would substantially decrease the time needed to integrate new devices to the system and keep the system easy to use.

A data integration system was developed for a laboratory environment containing four 2.0 l bioreactors (Biostat MD, B. Braun Biotech International, Germany) and one unit containing four 1.0 l bioreactors (Biostat Q, B. Braun Biotech International, Germany). All of these five units were connected to a commercial software Multi Fermentor Controlling System (MFCS/Win) version 2.1 (B. Braun Biotech International, Germany). The approach in this case was to use the MFCS as a node point for the system due to its capability of communicating directly with bioreactors and collect all primary process data in a practical database. The main challenge was to integrate all external measurement and laboratory devices needed in bioreactor cultivations to the system so that all data could be collected to the MFCS. There was also need for sharing online data with other applications like distributed controllers and monitoring applications.

3. INTERFACE MODULE

A microcontroller, ATMEGA8535 (Anon., 2006) based interface module capable of communicating with a PC using a standard serial port (RS232) was developed for devices using other than serial data interface (RS232), e.g. analogue signals. A simple single side circuit board was designed so that it could be used for multiple purposes without changing the layout. This feature was achieved by leaving a small raster board area to the other end of the circuit board, which can be used for the unique components required for a specific device module as shown in Figure 1.

Case specific software needed for each microcontroller was developed using C programming language (Programmer's notepad) and compiled using ARV-GCC. Programming of the microcontroller and debugging of the software was done by using AVRStudio 4.0 (Anon., 2006) and STK500 programming device (Anon., 2006).

Two different models of the modules were developed. The first model used RS485 signal (Figure 1) for PC end communication so that multiple modules could be connected to one serial port via active RS485-RS232-conventer. The second model was almost identical, but it used RS232 instead of RS485. This feature made it possible to connect any module directly to a PC serial port without an RS485-RS232-conventer, but it also supports connecting of multiple modules to the same serial port. In both cases a packet based protocol was used with Token Ring like network architecture so that all modules were able to push packets forward on the bus and also communicate with each other if connected together.

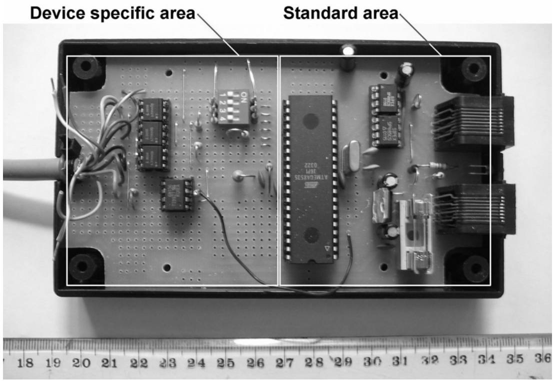

Fig 1. The microcontroller-based interface module contains a standard area and device specific area.

The real advantage of this approach was that almost any PC could be used for data acquisition because no special data acquisition card was needed to install to the PC. With a programmable microcontroller it was also easy to create a highly specific interface at the device end e.g. combination of analogue and digital signals and perform digital signal processing like taking mean values or calculating differences between signals.

4. DEVICE SPECIFIC SOFTWARE

Device specific software was developed for each laboratory device integrated to the system by using Visual Basic 6.0 (Microsoft, U.S.A.) at Windows 2000/XP environment. The software is able to communicate with the target device as required in each case. In most cases communication is done via serial port either directly or with the help of an interface module.

Although the physical interface (serial port) might be the same, there are almost as many protocols as there are devices. This is, however, only a programming challenge and many manufactures describe the proper protocol in the user manual. Depending on the device the software can be simple data acquisition software or more complicated, e.g. software taking care of an analyzer.

A common style for GUI (Graphical User Interface) was also established during the development of software versions. This helps users adapt faster to using new software, and also expedites software development time. GUIs are table formed and simplified containing options to select which bioreactor the device is related to and what are the latest results and events. The target variable, in other words the variable in MFCS, can also be changed from the GUI.

A common key feature for each program is the ability to read and write data to and from the MFCS via a software interface (later also via network interface). This feature makes data integration possible. Data transfer is in both cases packed based and upload and download intervals can vary depending of application. The MFCS has been configured in our case to work with minimum data acquisition interval of 15 sec and dead band zones to avoid saving unnecessary data. The instruments which we have integrated to the system are using, however, intervals varying from 1 second to 10 minutes. If primary data acquisition interval is less that 15 sec, we can upload the latest value every 15 seconds or take a mean value of the samples and send the filtered value to the MFCS. With longer intervals new data is uploaded only when a new value is received from the instrument. Different interfaces were also tested during development. The latest versions are using MFCSAPDA (Anon., 2002), but a commercial Visual Basic interface was also found functional. The most versatile interface would have been the DDE (Dynamic Data Exchange) interface, but it proved to be too unstable.

5. NETWORK FUNCTIONS

System flexibility was increased by developing a server application (mfcsServerPro) for MFCS capable of converting the software interface (MFCSAPDA) to a simple network TCP/IP interface. This implementation made it possible to share and collect online data via intranet and distribute measurement devices so that they could be connected to any PC in the laboratory intranet. As the device specific software is launched, it normally connects first to the device and then searches the route to the MFCS. If device is connected to the same computer where the MFCS is running, it uses the software interface, and if not, it connects to the MFCS via laboratory intranet and mfcsServerPro.

The network interface has also been tested in Matlab and Java environments with good results. There are no limitations for connecting third party applications to the system. The only requirement is the ability to establish a TCP/IP connection to the mfcsServerPro. A remote monitoring and control tool (Figure 2.) was developed so that cultivations could be monitored from any computer in the laboratory intranet. Integrating a GSM modem to the system made it possible to monitor cultivations from almost

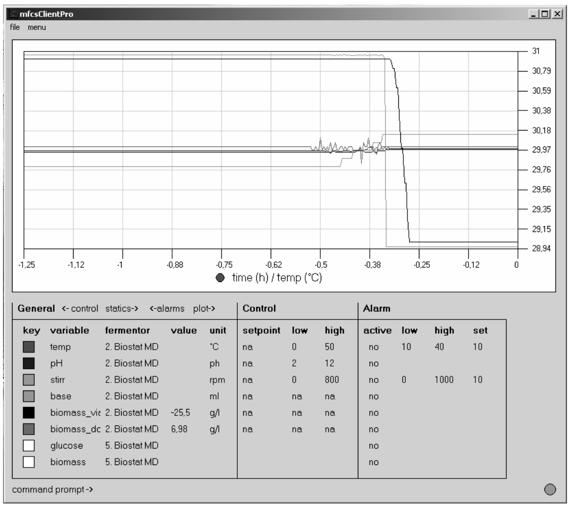

Fig. 2. User interface of remote monitoring and control tool mfcsClientPro.

anywhere by mobile phone and SMS. This also made it possible for mfcsClientPro to send SMS alarms to users if something unexpected was happening during the cultivations. This function can also implement to device specific software so that they can send SMS for operators, if acute maintenance is needed.

6. CURRENT SYSTEM AND FUTURE WORK

Currently more than ten different external laboratory devices are integrated to the system built around MFCS and eight bioreactors. Due to the system's

Table 1. Laboratory devices integrated to the system.

Device	Manufacturer	Description	Primary interface
Biomass Monitor 220	Aber Instruments Ltd	online, in situ viable biomass measurement	half duplex RS232
E-6000 with gas flow meters	Bronkhorst High-Tech B.V.	multi channel gas flow meter and controller	full duplex RS232
Sarorius EB35EDE-1	Sartorius	laboratory scale	half duplex RS232
Twist	Falcom	dual band GSM/GPRS modem	full duplex RS232
Hack Ratio/XR	Hack	turbidity meter	0-1 V analogue
TruCell	Finesse Instruments	online, in situ cell density measuring	4-20 mA
Watson & Marlow 101U/R	Watson & Marlow	peristaltic pump	0-5 V analog and logic level I/O
YSI 2730 Select	YSI Incorporated	at line biochemistry analyzer (glucose)	full duplex RS232
Omnistar & GSS300	Pfeiffer Vacuum	mass spectrometer with a gas stream selector	DDE-interface

modularity, only the devices required at a time are usually connected to the system and others can be connected when necessary. A list of the most important devices is presented in Table 1. Many other devices are also integrated to PC using the same techniques as described but are not integrated to the system, e.g. viscosity meter, pH meter and water bath.

Overview of data flow in the current system is illustrated in Figure 3. Although it's possible to send e.g. set point values from Matlab to laboratory devices, e.g. pumps due to integration, the MFCS is always the node point of the system. This means that the value is first written to MFCS and then the software controlling the pump reads it from MFCS and sends it to the pump. The only exception is the service for sending SMS, where the mfcsServerPro is the node point.

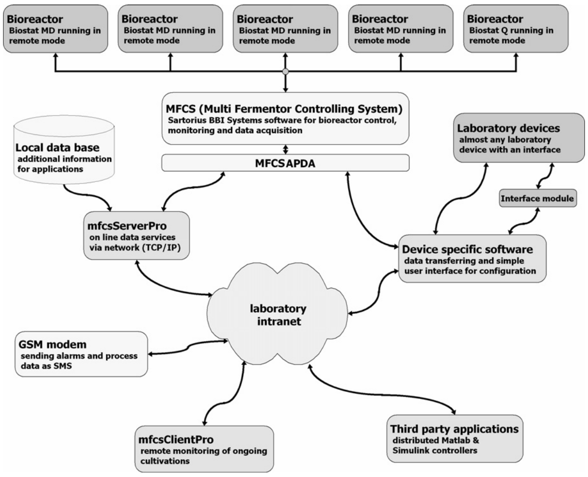

Fig 3. Data flow in the current system. All arrows present data flow and data can flow through any box if not terminal. Alternative routes are present parallel.

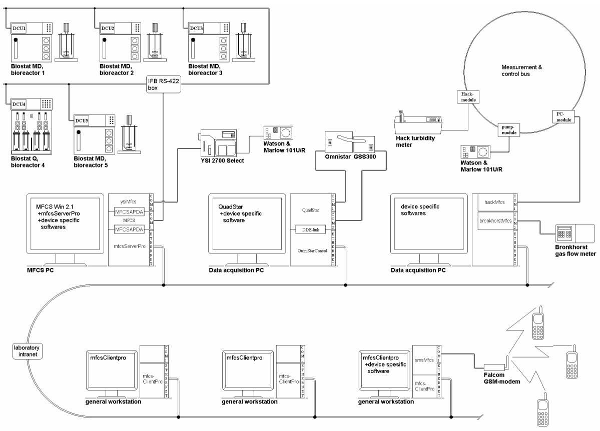

Fig. 4. Example of connecting laboratory devices. Network-based distributed data acquisition by using three computers and microcontroller-based interface modules.

The physical structure of the system is shown in Figure 4. The system structure presented in there is only an example.

It's also good to note that only about half of the currently integrated devices are presented in Figure 4. In the future it is also possible to integrate two 10 l and one 200 l bioreactors to the system for data acquisition. This is possible as bioreactors can be added to the MFCS and interface module and device specific software can be developed also for this purpose.

Bioreactor simulations in Matlab/Simulink-environment can be utilized to test different control schemes. A user interface shown in Figure 5 resembling a real control interface was created for teaching purposes.

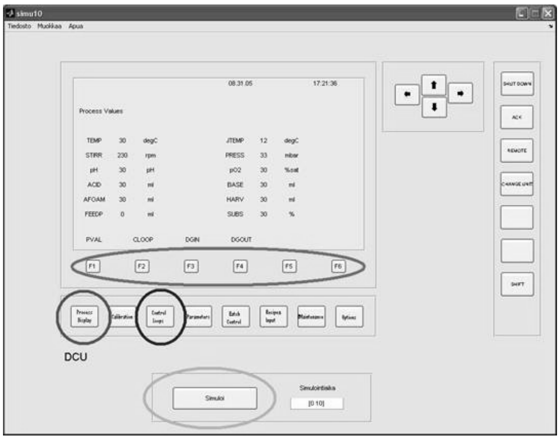

Fig. 5. Bioreactor user interface written for Matlab/Simulink environment

7. DISCUSSION

We developed a flexible network-based data acquisition system easy to append and capable of providing online process data for distributed applications. In our case the system was build around the MFCS, but principles described in this paper may also apply to almost any other software platform. The only demand for the large scale integration is an interface to the data acquisition software. If network based data acquisition is used, then only the server program (mfcsServerPro in our case) needs to be reconfigured. Although the system was limited only to the laboratory intranet, these advantages where archived well. In many cases this limitation is not made and systems are even more versatile (Bertocco *et al.*, 1998; Kumar et al., 2002; Tan *et al.*, 2002). The microcontroller-based interface module proved to be a simple and inexpensive method for integrating devices with analogue interface. These results and achieved features are common in microcontroller-based applications (Asimakopouls et al., 2005; Mukora and Carelse, 1999). The idea to use device specific software is quite obvious because all the devices are different and software is needed for every device. Visual Basic 6.0 was quite suitable for our purposes because designing of GUI was easy and serial port and network interfaces could be used very easily. Visual basic also offers basic low level commands such as bit operations and Windows API commands, which are sometimes needed. It’s also possible to use Microsoft Excel and its VBA language (Visual Basic for Applications) for a direct commication with MFCS. We didn’t need to use any commercial data acquisition cards where availability of drivers might limit programming language. If commercial hardware e.g. from National Instruments is used, then LabVIEW (National Instruments) might be a better choice for software development (Tan *et al*., 2002). All these kinds of systems are somehow unique due to different laboratory environments and devices. However, by using a combination of well-known methods, such as microcontroller-based data acquisition and network-based distributed measurements, it is possible to achieve even better results.

REFERENCES

Brown, F., M.G. Harris and A.N. Other (1994). Name of paper. In: *Name of book in italics or underlined* (Name(s) of editor(s). (Ed)), page numbers. Publisher, Place of publication.

Anon., 2002. B. Braun Biotech International GmbH,, MFCS/DA Programmer’s Guide, pp. 1-11.

Anon., 2006. Atmel Corporation, http://www.atmel.com/dyn/products/product_card.asp?part_id=2008

Asimakopoulos, P., Kaltsas, G. and Nassiopoulou, A.G. 2005. A microcontroller-based interface circuit for data acquisition and control of a micromechanical thermal flow sensor. J. of Phys. 10: 301-304.

Bertocco, M., Ferraris, F., Offelli, C. and Parvis, M. 1998. A client-server architecture for distributed measurement systems. IEEE Trans. on Instrum. and Meas. 47: 1143-1148.

Kumar, B.R, Sridharan, K. and Srinivasan, K. 2002. The design and development of a web-based data acquisition system. IEEE Trans. on Instrum. and Meas. 51: 427-432.

Mukora, R. and Carelse, X.F. 1999. A microcontroller-based data acquisition system for solar radiation and environmental monitoring. IEEE Trans. on Instrum. and Meas. 48: 1232-1238.

Tan, K.K., Lee, T.H and Soh, C.Y. 2002. Internet-based monitoring of distributed control system-an undergraduate experiment. IEEE Trans on Educ. 45: 128-134.