

A NEW CONCEPT IN GLASSHOUSE COMPUTER AUTOMATION WITH SCADA AND CASE TOOLS

W.Th.M. van Meurs, Th.H. Gieling and H.J.J. Janssen
Institute of Agricultural and Environmental Engineering (IMAG-DLO)
P.O Box 43
6700 AA Wageningen
The Netherlands

Keywords: climate control system, on-line database, fieldbus.

Abstract

Climate control computers in greenhouses control heating and ventilation, supply water, dilute and dispense nutrients and integrate models into an optimally controlled system. This paper describes how information technology, as in use in other sectors of industry, applies to greenhouse control. In other fields of industrial automation, the introduction of WINDOWS and OS2 gave way to the use of a whole new generation of hardware and software tools: fieldbus systems, a real time on-line database, computer aided program engineering and object oriented programming. The introduction of these modern concepts in horticulture adds extra power to climate control in greenhouses.

1. Introduction

A few years ago a new greenhouse was built at the IMAG-DLO site. This fact gave way to reconsider the general concept for hardware and software of process control in greenhouses. In the seventies IMAG-DLO was one of the pioneer institutes to introduce computer technology in greenhouses (Van Meurs, 1980). Then 'climate computers' were merely introduced as a digital replacement of the widely accepted continuous analog controllers. The actions of these controllers were copied in the software and the knobs were transformed in a keyboard (Gieling, 1980). Since then, new control techniques were made available to climate control in horticulture through research efforts at numerous places. Adaptive control was introduced (Udink ten Cate, 1983) when some of the processes taking part in the greenhouse climate showed a physically or mathematically ill-defined nature. The concept of using models in the control loop was described by Bot (1983) and Stanghellini and Van Meurs (1992). Another line was developed that of the "Speaking Plant Concept" which was described by Udink ten Cate et al. (1978) and Hashimoto and Morimoto (1984). Knowledge engineering techniques were introduced in climate control in protected cultivation by Hoshi and Kozai (1984). During the last few years the computer is not only used to control the greenhouse climate, but also to control the supply of water and nutrients (Gieling et al., 1994), and the integration of different models into an economically optimal controlled system (Van Henten, 1994; Van Meurs and Van Henten, 1994). These aspects ask for more advanced hardware and software and justified a new consideration wherewith new techniques make the research development faster and the introduction of results easier.

2. Starting points

In modern research it is a general rule to use special purpose models, when desk studies have to be performed and process scenarios have to be simulated. Only at the last stage,

experiments are performed in reality. Modern research in greenhouses shows no difference to this general rule. Models are developed and applied on-line to the controlled process: mathematical models to support the control and system engineer, physical and physiological models to support the physicist and horticulturist. They all expect that a Fortran-like simulation language is supported and that their models are directly and easily put on-line with their own experiments. They expect no interference from the experiments in other greenhouses. Measurements are performed during tests and the resulting data are expected to be available on-line with the models and to be stored for off-line data processing. Researchers expect that their program, processes and data can be monitored from their own desk-top work-station in their own office through a local area network (LAN) connection. On the other hand, the system and maintenance groups expect an open system with a minimum of intervention and explicit tools for fast analysis of problems and solutions.

A system comprising all these options is depicted in Figure 1 and shows four levels of operation.

Level 1: the top of the hierarchical structure is a mainframe computer connected to the 'outside world' by the Ethernet (Local Area Network). The physical and software layers of the Ethernet consist of hardware interfaces, a software protocol and at least one server. The LAN interconnects mainframe, desktops, workstations, cluster computers, printers, plotters and hard disks.

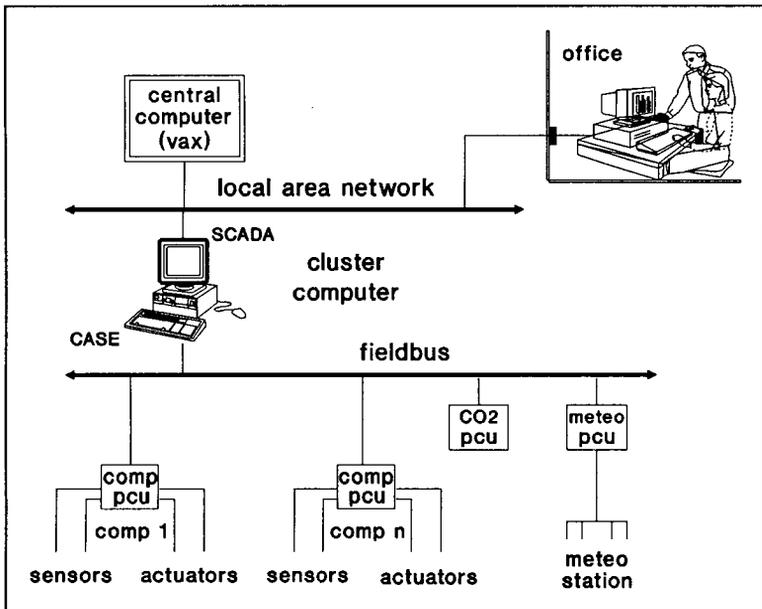


Figure 1 Four operational levels of the computer system

Level 2: a desktop PC or workstation connected to the LAN offers the opportunity to the researcher to work at his desk, for program development, changing set points, data acquisition and data processing.

Level 3: the cluster computer manages the data of a clustered group of local Process Control Units (PCUs).

Level 4: a local Process Control Unit (PCU) acquires sensor data and controls actuators. Process control is completely performed at this lowest hierarchical level. All standard control algorithms perform their function at this level. The PCUs are clustered in groups. They interconnect by means of a fieldbus, headed by their cluster computer as bus-master. Each compartment has his own PCU.

3. Materials and Methods

3.1. Cluster-computer

The cluster computer integrates tools for on-line and real-time process control and data management. OS2 was selected as the basic platform because of its multi-tasking capabilities and window-like environment, however without the shortcomings of these popular environments. OS2 is a robust operating system: a crash of a task does not result in a crash of the whole system.

This computer uses a Supervisory Control And Data Acquisition (SCADA) package. Examples of commercial SCADA packages in industrial automation are ONSPEC, FIX-DMACS, FactoryLink, GENESIS, CONTROL VIEW and WIZCON. Most of these packages incorporate tools like Computer Aided Software Engineering (CASE), Object Oriented Programming (OOP), knowledge base techniques, PID-, fuzzy- and neural network-based control algorithms. The SCADA software, basically a real-time on-line database, that was selected for this task at the IMAG-DLO is FactoryLink from the firm USDATA (USA) and is noteworthy for its open structure. Commercially available SCADA software use standardized protocols to enable an open structure of the database for hardware and software from outside.

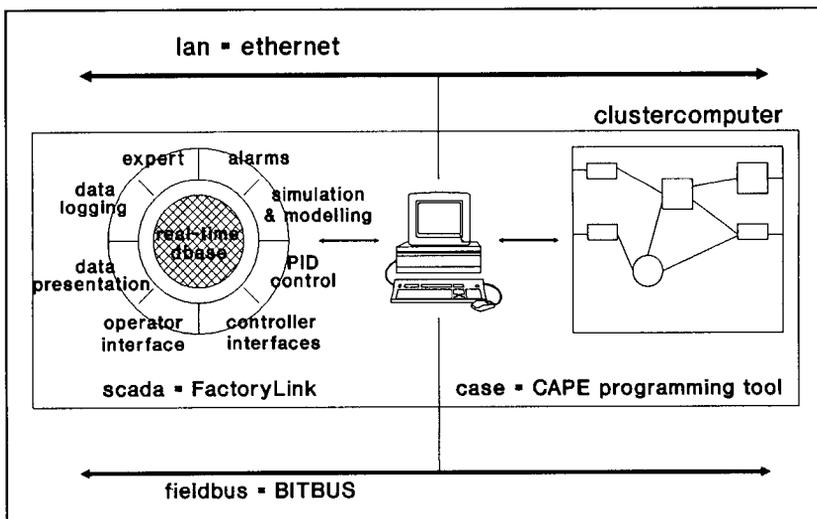


Figure 2 The cluster computer

A large number of software toolboxes and hardware equipment can be easily coupled to it. A number of tasks have been added to this database (Figure 2): the graphical user interface (GUI) simplifies operator actions with animated process drawings, diagrams and pictures, data acquisition and processing, graphic representation of process data, and alarm handling. These features guarantee a low entry threshold for users like the researcher, programmer, operator, maintenance staff and the day-to-day supervisor of the glasshouses. Each user can have an authorized access to a predefined set of parameters in the database, in order to prevent accidental interference or unauthorized changes of set points. Data can come from sensors, from manual input, from a knowledge base, from a database or from on-line calculations. Parts of the database on the cluster computer can be shared via the LAN with databases on mainframes or workstations.

A CASE program called CAPE was selected as a software development tool for generating control instructions for the PCUs. CAPE has been developed by Technolution, The Netherlands. The CAPE tool consists of a set of graphical objects called 'atoms'. Each atom owns input and output data fields. Input and output data fields from different atoms can be interconnected graphically by a line, the data path, which can be drawn with the mouse. The sequence of handling of atom operations is along a second line interconnecting the atoms. Atoms represent standard operators from information theory (or, and, xor, not, multiply, divide, add, <, >, =, subtract, sequence-switch, data-switch, etc.), storage elements (variable, constant, array etc.) and functions (smooth, sum array, PI-control etc.). When a screen is filled with objects it can be minimized into a new object, the 'molecule'. The 'variable' storage elements in the original screen are the input or output data fields of the molecule. Encapsulation is an important property of object oriented programming. In CAPE it is guaranteed by special objects. During testing, it is possible to scroll on-line through the program of atoms and molecules. The data-fields of the graphic objects remain connected with the on-line process and display continuously the actual values of the process in the greenhouse. So, the connection between monitoring and control hardware, the bus I/O and the database elements remains intact. This feature adds a unique dimension to the debugging of the design. For reasons of maintenance it was decided to use commercially available software.

3.2. The fieldbus

The choice for a fieldbus for greenhouse climate control was determined by the following criteria:

- an open system to meet international industry standards,
- a bus topology with a working range of at least 1 km,
- a robust industrial design, easy to install, easy to expand and easy to service,
- a complete galvanic separation of fieldbus and computers for protection against interferences, especially caused by lightning,
- the two bottom layers of the OSI model must be carried out entirely by the network (built-in protocols).

An American, a European and a Japanese fieldbus standard is in preparation. The industry has developed a range of standards with a distinct national flavour such as the English MIL-bus, the German Profibus and DIN Messbus and the French FIP. BITBUS, developed by INTEL, was selected for this application, because it meets most of the standards and has been provisionally approved by ISO. A general ISO standard for the fieldbus is unlikely to be expected before 1995, but BITBUS has been approved under IEEE 1118-1990. BITBUS is based on the RS485 standard and the SDLC protocol. It is a master/slave protocol incorporated in hardware, with a known response time (deterministic). A transmission speed of 375 Kbytes/s was selected for this project, because of the allowed bus length of 1 km. The

BITBUS protocol is handled by a dedicated processor in the cluster computer as well as the PCU. The cluster computer uses a PC interface board to handle communications, which supports all BITBUS functionality. The firm INCAA Computers (the Netherlands) developed the PCU hardware and driver software to interface FactoryLink and CAPE to BITBUS.

3.3. The Process Control Unit

A permanent multitasking core of up to 7 user tasks is contained in a 8044 processor. The multi-tasking operating system was incorporated in ROM. The user tasks implemented consist of PCU-Cluster communications (BITBUS protocol), communications with CAPE, global exchange of data between PCUs, alarm handling (Figure 3). One of the tasks is the communication between different hardware boards inside PCU via a I²C-bus. A 80c188 processor runs the whole control process. The control algorithms are implemented in CAPE objects.

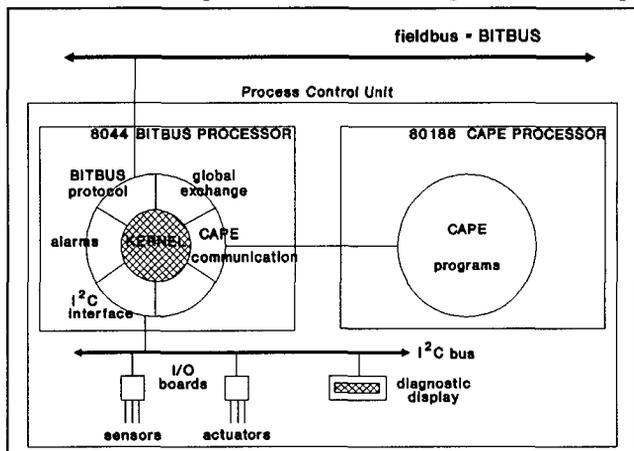


Figure 3 The Process Control Unit

At system start-up, the programs are down-loaded from the cluster computer to the PCU via the fieldbus and started. Measurements of sensors and the control of output relays is performed using CAPE objects on different boards. Sensors and actuators are controlled by separate boards, each with its own processor. The signal conditioning, counters, timers, periodic calibration and test procedures are all entirely self-contained on these I/O boards. A diagnostic display indicates the status of the PCU (on-/off-line, alarms).

3.4. Climate Control

The climate control module in the PCU consists of basic building blocks for climate control, which are created by putting together the atoms and molecules of CAPE. These basic building blocks calculate set points and actually control humidity and temperature of the greenhouse air. They control the position of actuators like windows, heating valves, energy saving screens and shading screens. Additionally the supply of CO₂, the application of sulphur and the on - off time of artificial lighting are controlled.

The basic climate control algorithm is divided into software modules for e.g. heating, ventilation and CO₂ supply. These modules contain sub-modules for set point calculation and P(ID) control. The (sub)module receives inputs from the on-line measurements, and

parameters from the DBASE. It produces outputs for steering the relays and parameters that can be stored in the DBASE. Each (sub)modules starts with a software switch.

There are two ways to implement a new- or redesigned module. The first one is to develop or rewrite the module in the CASE development toolkit with 'atoms' and 'molecules' and start the new module for a specific compartment (Figure 4a). The PCU software identifies a new version of the program. Automatically the old version stops and the new version is loaded in the PCU and will be executed.

The second method is to execute the new module (written in Fortran, C or any other computer language) as a program, directly in the clustercomputer (Figure 4b). This program has to be supplied with input and output parameter (tagnames) of the DBASE. One of the output parameters controls the software switch of the (sub)module to be superseded. The output values of the program, put in the DBASE, are immediately sent to the PCU.

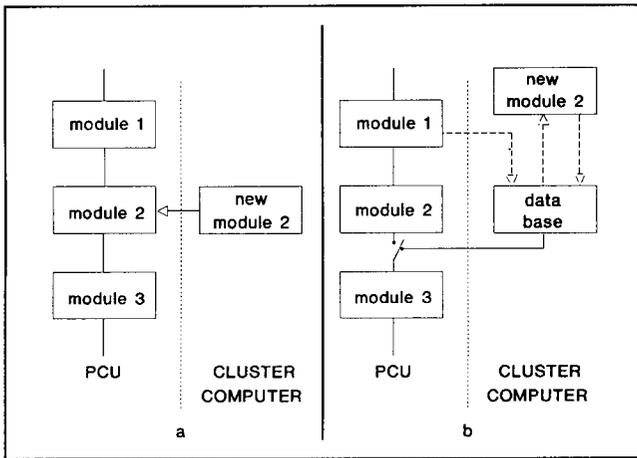


Figure 4 Implementation of new modules

4. Conclusion

The state of the art of information technology used in climate control in greenhouses, was lagging behind with respect to developments in other sectors of industry. The introduction of new operating systems like WINDOWS for popular limited multi-tasking computer applications and OS2 for enhanced multi-tasking industrial applications, gave way to the industrial use of a whole new generation of software technologies and tools. Until now these tools and techniques were not at all used in horticultural practice and hardly used in horticultural research. So, a complete overhaul was more than overdue.

References

- Bot, G.P.A., 1983. Greenhouse climate: From physical processes to a dynamic model, Ph.D. thesis Agr. Univ., Wageningen: p 270.
- Gielsing, Th.H., 1980. Commercial greenhouse computer systems. *Acta Horticulturae* 106: 59-66.
- Gielsing, Th.H., Bontsema, J. and Van Antwerpen, A.W.J., 1994. Dynamic behaviour of salinity changes in closed growing systems for plant production. *Submit: Advances in Space Research*, Pergamon Press, Oxford.
- Hashimoto, Y. and Morimoto, T., 1984. System identification of plant response for optimal cultivation in greenhouses. *Proc: 9th IFAC World Congress*, Vol.7:2039-2044, Pergamon Press, Oxford.
- Hoshi, T., Kozai, T., 1984. Knowledge-based and hierarchically distributed online control system for greenhouse management. *Acta Horticulturae* 148: 301-303.
- Stanghellini, C. and Van Meurs W.Th.M., 1992. Environmental control of greenhouse crop transpiration. *J. Agric.Engng Res.* 51: 297-311.
- Udink ten Cate, A.J., 1983. Modelling and (adaptive) control of greenhouse climate, Ph.D. thesis. Agr. Univ., Wageningen: p 159.
- Udink ten Cate, A.J., Bot, G.P.A. and Van Dixhoorn, J., 1978. Computer control of greenhouse climate. *Acta Horticulture* 87: 265-272.
- Van Henten, E.J., 1994. Greenhouse climate management: an optimal control approach. Ph.D. thesis. Agr. Univ., Wageningen: p 329.
- Van Meurs, W.Th.M., 1980. The climate control computersystem at the IMAG, Wageningen. *Acta Horticulturae* 106: 77-84.
- Van Meurs, W.Th.M. and Van Henten E.J., 1994. An experiment on the optimization of CO₂ in greenhouse climate control. *Acta Horticulturae* 366: 201-208.