2002 NASA/ASEE SUMMER FACULTY FELLOWSHIP PROGRAM

JOHN F. KENNEDY SPACE CENTER
UNIVERSITY OF CENTRAL FLORIDA

IMPLEMENTATION OF AUTONOMOUS CONTROL TECHNOLOGY FOR PLANT GROWTH CHAMBERS

Thomas A. Costello
Associate Professor
Biological and Agricultural Engineering Department
University of Arkansas

John C. Sager
Agricultural Engineer, NASA-KSC

Valdis Krumins
Biological Engineer, Dynamac Corp.

Raymond M. Wheeler
Plant Physiologist, NASA-KSC

ABSTRACT

The Kennedy Space Center has significant infrastructure for research using controlled environment plant growth chambers. Such research supports development of bioregenerative life support technology for long-term space missions. Most of the existing chambers in Hangar L and Little L will be moved to the new Space Experiment Research and Processing Laboratory (SERPL) in the summer of 2003. The impending move has created an opportunity to update the control system technologies to allow for greater flexibility, less labor for set-up and maintenance, better diagnostics, better reliability and easier data retrieval. Part of these improvements can be realized using hardware which communicates through an ethernet connection to a central computer for supervisory control but can be operated independently of the computer during routine run-time. Both the hardware and software functionality of an envisioned system were tested on a prototype plant growth chamber (CEC-4) in Hangar L. Based upon these tests, recommendations for hardware and software selection and system design for implementation in SERPL are included.

IMPLEMENTATION OF AUTONOMOUS CONTROL TECHNOLOGY FOR PLANT GROWTH CHAMBERS

Thomas A. Costello
John C. Sager
Valdis Krumins
Raymond M. Wheeler

1. INTRODUCTION

1.1 Background

Human space missions require facilities for ongoing supply of clean water, oxygen and food (and the removal of carbon dioxide). Although these resources can be Earth supplied on short duration and/or short distance missions, the high cost of re-supply during longer, more distant missions will require other options such as bio-regeneration or manufacture from local planetary material. Bioregeneration involves use of living systems (crop plants, algae, small animals, microbes) to provide net photosynthesis to maintain oxygen and carbon dioxide balance for the crew, as well as providing clean water, food and nutrient recycling to offset needs for resupply. Wheeler (1996) estimated that plant growth chamber facilities having 40 m^2 of planted crop per crew member could provide major food requirements and provide oxygen for human needs as well as oxidation of all organic waste (and potential for nutrient recycling). Drysdale and Finger (1997) provided a conceptual basis for considering bioregeneration with only partial closure of the food, oxidation and recycling loops. Decisions regarding overall system design and the need and cost of re-supply can be analyzed using an equivalent system mass approach (Levri et al., 2000).

Continued research is needed to quantify the mass balances of food crop production grown in enclosed chambers to simulate long-term space missions. The efficiency and rate of biomass accumulation and the fraction of biomass that is edible will directly impact the size of bioregenerative systems and the need and cost of partial resupply from Earth or local planetary resources. Methods of nutrient recovery and re-use need to be tested as well to identify microbes and plants which optimize these processes (e.g., Garland et al., 1993; Mackowiak et al., 1996).

On-going research in bioregenerative life support involves operating controlled biological reactors for plant growth, aerobic solids decomposition (in-vessel composting), aerobic liquid waste decomposition (membrane and other bioreactors) and other specialized systems. In such systems, not only is it necessary to control environmental process variables in order to express reproducible treatment effects, but often the environmental variables are targetted at levels intended to simulate some space-based system (shuttle, space station, planetary outpost). For example, light spectral quality and flux density as well as atmospheric pressure may be adjusted to investigate possible plant growth characteristics in a Martian greenhouse environment.

Biologists and engineers who develop bioregenerative life support experiments may be constrained by the level of complexity required to configure, control and monitor living systems in the laboratory. Hardware and software are used to process electronic signals from various sensors, compare existing conditions to some predefined standard and then make decisions to adjust/operate/deactivate various control devices (such as pumps, fans, valves, relays, etc.). In the past, these functions required significant design effort from a biological engineer and on-going support from an instrument technician. Newer hardware and software options can decrease the technical overhead so that an engineer and technician can provide

support for more experiments and investigators can take a more direct role in designing an experimental methodology and subsequent data recording. More powerful, customized software may empower a scientist to conceive and implement an experimental protocol that was previously passed over due to technical constraints. A new generation of control and data recording hardware and software has the potential to decrease costs, increase reliability, improve information quality and potentially lead to new discoveries.

At NASA-KSC, the Bioregenerative Life Support research programs are operated as part of the Biological Sciences Office under the Spaceport Technology Center. Existing plant growth chambers (and other experimental biological systems) function under the direct control of a central work station computer, using NASA-written custom software (Universal Data Acquisition and Control Engine, UNDACE). The established procedures for replication of routine experiments using UNDACE operate efficiently and reliably. However, the intellectual property that UNDACE represents is rapidly becoming devalued because of attrition of technical staff who had the specialized knowledge required to maintain the software and hardware. New configurations and control suites are not attempted due to this uncertainty. The expected move of the growth chamber complex to SERPL in 2003 underscores the need to replace UNDACE and the associated hardware with an alternative system which requires less specialized technical support that can be maintained over the next decade of biological research.

Concepts for an improved control and data recording methodology include the following features:

> 1) Use of autonomous control modules which do not require continuous interaction with the supervisory computer (in order to better maintain control during periods of disrupted communication).
> 2) Backup control capability existant in commercial chambers, which can take over to prevent catastrophic experiment failure (death of plants or microbes in a reactor) should the customized, external control fail.
> 3) Communications using ethernet connections to provide simplicity in wiring and extend possibilities for better process monitoring by investigators from any workstation.
> 4) Capturing and making real time data available for web-based applications using XML or other technologies which are platform independent. These capabilities will increase the efficiency of data analysis and increase the extraction of knowledge from the research. It will also promote potential utilization of research results by clientele outside NASA, including potential educational web-sites for K-12 students interested in following on-going space-based biological experiments.
> 6) Providing a suite of software tools that allow rapid configuration and interfacing between chamber equipment and control modules so that a biologist with minimal technical support can devise and implement innovative experimental protocols.
> 7) Supplementing the graphical interface associated with UNDACE with improved visual displays that promote on-going human observations of experimental conditions. Better visualization means more power to detect irregularities associated with equipment malfunctions. Moreover, scientists often extend their systemic knowledge when interesting phenomena become visible. There is real potential to pull more knowledge from each experiment using an improved human-machine interface.

Specifically, a proposed strategy would include the use of programmable logic controllers (PLC's; model SNAP-B3000-ENET and associated modules, OPTO 22, Temecula CA) connected through an ethernet connection to a PC running supervisory software developed using a graphical programming language (LabVIEW, National Instruments, Austin, TX). The PLC's can operate independently with periodic

communication with the PC for data recording and to provide the human-machine interface. The SNAP hardware had been used to control and monitor other experiments in Hangar L, but the specific module needed for temperature control in a plant growth chamber (SNAP PID Module) and interfacing using LabVIEW had not been attempted.

1.2 Objective
The objective of this research was to test the functionality of SNAP-B3000-ENET hardware and a LabVIEW-based user-interface for controlling and monitoring advanced life support experiments, specifically by testing a protoype system for a plant growth chamber. Following development and testing, results and recommendations will be made to NASA regarding future system development for the SERPL facilities.

2. METHODS

2.1 Hardware System Prototype
The prototype growth chamber (Figure 1) was located in Room 210 of Hangar L at KSC and designated 'Controlled Environment Chamber 4' (CEC4). The chamber (model M-12, EGC, Inc., Chagrin Falls, OH) is a reach-in chamber having a growing area of 1.4 m x 0.9 m to accommodate 8 hydroponic trays, each 80 cm x 13 cm. The chamber is currently set up to handle nutrients solutions for 4 treatments separately. Nutrient solution pH is monitored using pH probes and controlled through the addition of acid (HNO_3) using commercial pH controllers (model PHCN-37, Omega Engineering, Stamford, CT). There are 3 lighting banks that can be selected: "1/3FL" consisting of 7 fluorescent tubes (F48/T12/VHO/CWF), "2/3FL" consisting of 13 of the same tubes, and "HPS" consisting of 4 high pressure sodium lamps (400 W, G.E. Lucalux). The banks can be used individually or in combination on a selected diurnal time schedule. Current studies have used the HPS lights only with a typical photoperiod of 18 hours of light per day with an average intensity of 350 μmol m^{-2} s^{-1} PPF. An integrated heating/cooling refrigeration unit was used to circulate conditioned air through the chamber to maintain the desired chamber air temperature, with supply air temperature regulated by a hot gas bypass valve (Barber-Coleman) which was actuated with a 6-9 VDC analog control signal. Separate units for dehumidification and humidification were actuated using on/off relay control as needed to maintain a desired humidity level. The CO_2 concentration in the chamber was monitored and controlled to a target level using the existing UNDACE system.

The integral controller (model TC2) provided by the manufacturer of CEC4 uses a proprietary algorithm for proportional-integral-derivative (PID) control and usually controls chamber temperature with a ripple of about 0.3 °C amplitude. External control using UNDACE (and later using SNAP-B3000-ENET or "PLC" modules) was effected using a bank of double-throw relays to re-route control signals to the equipment via either the EGC controller or the PLC modules. A selector switch was installed to allow the operators to manually revert to EGC control if the computer or PLC's needed servicing. The wiring panel (Figure 2) used with the PLC's housed the power supply (24 VDC), thermocouple transmitter (to convert thermocouple output to a linear 2-10 VDC output roughly calibrated to a temperature of 0-100 °C) and a bus containing the individual PLC modules. For CEC4, we used the SNAP-B3000-ENET brain module, two SNAP-ODC-SNK digital output modules, two SNAP-AIV-4 analog input modules, and one SNAP-PID module. The wiring panel was connected to the chamber using multiconductor cable terminated with DB-25 connectors. A cable drop (Category 5, 4-twisted pair) was connected to the brain for ethernet communications. All of the control equipment was powered through an uninterruptable power supply. The chamber equipment was wired on a special panel which coulld be powered by emergency generator if

local power was interrupted.

After installation of the SNAP hardware and connection of sensors and control wires to the chamber, the parameters of the control algorithm PID module were adjusted to attain the 'best' control response. The adjustment procedure, called 'tuning' followed the so-called Ziegler-Nichols Closed Loop Method which involves a directed sequence of trials in which parameters were varied, response was observed, measurements made of the response and then a new set of parameters was tried. Due to poor documentation from the PID module manufacturer, this process was tedious and time consuming (required about 60 hours).

2.2 Software System Prototype

A limited user-interface for the CEC4 was programmed using LabVIEW (Full Development System). The functions that were included in the interface included: a) real time visual display of data (individual digital and analog indicators for key variables, banks of switches and lights for digital outputs and logical states), b) a time-moving strip chart graph that can display multiple traces (for multiple variables) on the chart, and c) dialog boxes to display/change/update the various setpoints for the control logic. The software product, a virtual instrument (VI) named MainOPTOMonitorChange.VI called several sub-VI's, as shown in Figure 3. The strip chart recorder display and other indicators were configured from control elements included in the LabVIEW package, as were various dialog boxes associated with the main VI.

Communication between the VI and the PLC's was accomplished using two methods. To get data, a VI was configured to poll an XML data file that had been uploaded to and serviced by the PLC brain module. Individual data items were extracted using an XML parser (an Active-X class of procedures named MSXML2.IXMLDOMDocument2, available free from microsoft.com). Unfortunately, the existing PLC firmware did not support memory-write capability using XML, so parameter updates were written to the brain's memory using another Active-X class, named OptoSnapIoMemMapXLib.IO22SnapIoMemMapX, which was supplied by the PLC manufacturer.

Due to time constraints, it was not feasible to attempt to program functions for configuring the PLC's and to record data. These functions were accomplished using existing OPTO-22 configuration web-pages and an exisitng custom data recording program (written using Visual Basic, collected data from the brain using a streaming data socket and stored data in a text file), respectively. Additional capabilities are still required to implement a fully automatic supervisory control, including error-checking, alarming, and integrated achival of quality-checked data.

3. RESULTS AND DISCUSSION

3.1 Autonomous Control

In general, results from these tests, combined with earlier experience using the OPTO SNAP-B3000-ENET in other experiments in Hangar L, indicated that the hardware was robust in terms of extended periods of autonomous operation in the laboratory environment, with modules operating as expected. However, the PID module performance was disappointing.

Under nominal conditions, UNDACE and the EGC controller were able to control chamber temperature with a deviation of about $\pm$ 0.5 °C and $\pm$ 0.3 °C, respectively. With the OPTO-SNAP-PID module, control was more variable sometimes ranging $\pm$ 0.5 °C and then unexpectedly shifting to a different waveform having a range of $\pm$ 1.0 °C, see Figure 4. Other times, the OPTO module was observed to control at $\pm$ 0.2 °C for a short period of time (data not shown), but this desired behavior would disappear

following some routine perturbation (like lights coming on or the door being opened).

It is not clear that the tuning procedure attained the optimal set of PID parameters. The module documentation and accompanying software tools contained conflicting terminology, units, errors and a great deal of ambiguity. A normally straight-forward tuning procedure actually became more of a trial and error procedure, because of these uncertainties, with infinite possible combinations of the PID parameters potentially to test. There is a major concern about the reliability of the module's internal program, particularly relating to implementation of the derivative control parameter in the PID algorithm. These concerns were shared by another NASA engineer who tested the module independently and urged caution about using the SNAP-PID module (Curtis Ihlefeld, electrical engineer, NASA-KSC, personal communication, July, 2002). These deficiencies seemed to be purely analytic (involving the internally programmed logic, documentation and interface software) and did not seem to involve hardware/instrumentation; therefore, it is possible that a future firmware/software upgrade could provide a remedy.

The SNAP-B3000-ENET brain uses Digital Events, Alarms and Timers as formal structures to define the control logic. Defining, coding and documenting logic using these structures is very tedious and prone to error, which leads to the need for significant debugging. In general, the SNAP system provided flexibility to perform the desired control functions. However, there is no provision for control events to be triggered by PC clock-time (events can be triggered only by elapsed time referenced to some previous event). This requires a tedious process to synchronize control cycles to clock time and human work schedules. This can be done manually on a periodic basis or it could be programmed to occur automatically within the supervisory software.

3.2 Supervisory Software
A limited software system that provides the human-machine interface was developed using LabVIEW. Analog, digital and strip chart displays were developed which provide adequate visual representation of the chamber processes to a scientist, engineer or technician. A sample of the real time display is shown in Figure 5. The displays utilize visual control components from the LabVIEW package and provide a distinct improvement in visual quality of the display, compared to the UNDACE system. These controls are easy to include in an interface; however, customizing the controls is difficult because most object properties are poorly documented in LabVIEW. An experienced LabVIEW programmer will gradually learn to adjust the controls as needed, but only through significant investment in a trial and error process. This may not be an issue if the interface is designed and maintained by an experienced LabVIEW programmer; however, if future modifications are intended to be made by in-house technical staff then special training may be required.

The interface was developed in about 80 hours by an engineer with considerable conventional programming experience (including Visual Basic) and some prior exposure to another graphical programming environment (but with no direct LabVIEW experience). Understanding the basic LabVIEW scheme and implementing simple single-process programs (or virtual instruments, VI's) was accomplished fairly quickly. Programming a more complex system (with multiple sub-VI's) was tedious and frustrating due to poor documentation of methods and properties. The system includes multiple examples but they are not linked directly to the context-sensitive help. Progress in development often occurred by guessing how it might be accomplished and then trying multiple alternatives until one actually worked.

The LabVIEW system seems particularly unsuited to building a user-interface for on-going

monitoring/data-recording of a process. Since processing in LabVIEW is data driven, and there can be multiple threads of code running in parallel, it is difficult to coordinate and respond to user input with predictable and satisfactory results. For example, a menu editor can be used to develop a menu structure, but the events triggered as the user navigates the menu are difficult to control. Sub-VI's that pop-up in response to mouse-clicks won't always stay or disappear as needed. Clicking a button in a sub-VI may not be processed in a sequence as expected. Again, a more experienced LabVIEW programmer may be able to work this out, but it is not a natural consequence of the LabVIEW architecture. A programmer using Visual C or Visual Basic could accomplish the functionality of responding to user mouse-clicks much quicker and more reliably.

The LabVIEW interface worked well in interacting with external program libraries (Active-X classes of sub-programs). Establishing a reference to the class, performing methods and changing properties were straight-forward. The LabVIEW documentation suggested that VI's could be compiled, packaged as Active-X components, and able to be called by other software. This was not tested because the Full Development package did not support creating a stand-alone file version of a VI. Assuming this was possible, one might use Visaul Basic to develop the user-response framework (menus and dialog boxes) to call LabVIEW VI's to display data within various windows on the screen.

Because of the extended development time required for the real-time display and parameter change functions, other functionality that should be included in a supervisory control package was not attempted. The functions not tested included: a) the process of configuring PLC modules and control regimes for a new experiment, b) the ability to define and implement a data recording session, c) the process of checking data for errors in case of failure in the chamber equipment or PLC's, and d) a method for archiving error-checked data in a format easily accessible by the scientists.

4. CONCLUSIONS

General comments regarding the performance and suitability of both OPTO-SNAP hardware and LabVIEW for this application, based on these tests, were listed in the discussion above. Specific recommendations for monitoring/control/data-recording in SERPL are listed here.

> 1) Further testing is needed to insure that the OPTO-SNAP-PID Module is a reliable controller for experiments in which cost of temperature control failure is high. Efforts should be made to work with the manufacturer to establish the accuracy of the PID algorithm. Alternatively, another PLC manufacturer may be sought, but this would necessitate movement to a completely new hardware system which would negate the expertise already established within Hangar L staff in supporting the OPTO modules. In its present configuration, the OPTO-PID module is not recommended.
> 2) Testing of prototype systems is needed to apply PID modules to the control of CO_2 and humidity. The CO_2 control actuation could be identical to that used at present with UNDACE with gas valves controlled by a PID module in time-proportional mode. Time proportional control could also work to control humidity. The current control algorithm for humidity is a simple on/off control with set points and an arbitrary on-time and subsequent wait-time. Time-proportional control would be more stable. In addition, because of the impact of the dehumidification processes on the sensible heat of the air mass, and because sensible heating lowers the relative humidity, the controls for air temperature and relative humidity appear to sometimes work against each other. A more stable control for humidity might be obtained using

a set point for absolute humidity (or vapor pressure) associated with the set point temperature, rather than targeting a specific relative humidity. This would require calculation of the vapor pressure based on dry bulb temperature and relative humidity (or by selecting a different sensor which provides a direct indicator of absolute humidity, such as a dew point hygrometer). At present, there is no direct way to use the SNAP hardware to make these calculations in an autonomous mode to provide a dynamic measure of the process variable (vapor pressure). Alternate hardware might have the capability to perform user-specified and uploaded calculations within the PID algorithm.

3) It is recommended that OPTO upgrades be sought that will provide support of XML communications with the brain in both the read and write modes. This would simplify the communications and increase the access to the data and machines in the future.

4) A dedicated work station will be needed to provide continuous supervisory control and to record and archive data. For best operator performance, it is suggested that a large screen be installed to provide real time display, located near the chambers themselves. Because of the ethernet connection, investigators can monitor and adjust processes from other locations on the network by running the software on a local PC.

5) User interface design should build on the successful results using UNDACE. Users are generally satisfied with the interface. Changes should be carefully considered to insure that an improvement is actually made. Changes that increase the flexibility and ease of use without sacrificing any current functions should be sought.

6) Data recording processes should include not only the dynamic chamber variables, but should also provide a log of the current control parameters and indicate when changes are made.

7) There is a need to develop a formal protocol for sensor maintenance and calibration, with appropriate staff assignments and budget to get this done.

8) It is suggested that a comprehensive documentation library be developed, including inventories of equipment, wiring diagrams, software source code (with comments) operating procedures, trouble-shooting guides and maintenance schedules and procedures.

9) It is recommended that LabVIEW product upgrades or alternative software be selected for development of the permanent user-interface. LabVIEW seems to be valuable for developing screen displays, but other programs might be more efficient in controlling user interaction (menus, dialog boxes). Experienced consultants may provide guidance for the most efficient combination of software development tools.

10) Permanent user-interface design should be preceded by a detailed definition of the flexibility needed by users in configuring future systems, in terms of new hardware, new control schemes and new data recording schedules. It should be decided if the interface will provide for these changes or if new source code will be required. Also, it should be defined how much technical expertise will be required to make future modifications to start a new suite of experiments. These decisions will impact greatly the architecture of the new software.

REFERENCES

[1] Wheeler, Raymond M. Gas balance in plant-based CELSS. In: H. Suge (ed.) Plants in Space Biology, pp. 207-216. Institute of Genetic Ecology, Tohoku University. 1996.

[2] Drysdale, Alan E. and Barry W. Finger. Waste processing for advanced life support:influences on operational strategies and design. SAE Technical Paper No. 972292. Soc. Automotive Eng., Warrendale, PA, USA. 1997.

[3] Levri, Julie A., David A. Vaccari and Alan E. Drysdale. Theory and application of the equivalent system mass metric. SAE Technical Paper 2000-01-2395. Soc. Automotive Eng., Warrendale, PA, USA. 2000.

[4] Garland, Jay, Cheryl L. Mackowiak and John C. Sager. Hydroponic crop production using recycled nutrients from inedible crop residues. SAE Technical Paper No. 932173. Soc. Automotive Eng., Warrendale, PA, USA. 1993.

[5] Mackowiak, C. L., J. L. Garland, R. F. Strayer, B. W. Finger and R. M. Wheeler. Comparison of aerobically-treated and untreated crop residue as a source of recycled nutrients in a recirculating system. Adv. Space Res. 18(1/2):281-287. 1996.

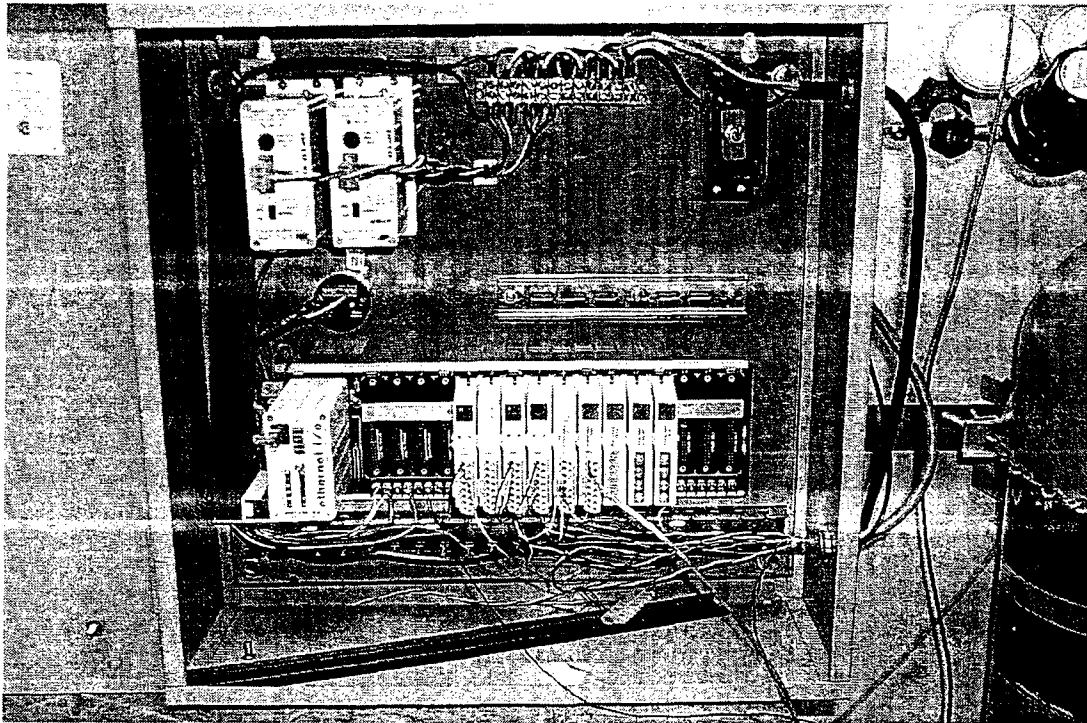Figure 1. Plant growth chamber used to perform testing of the prototype control system.

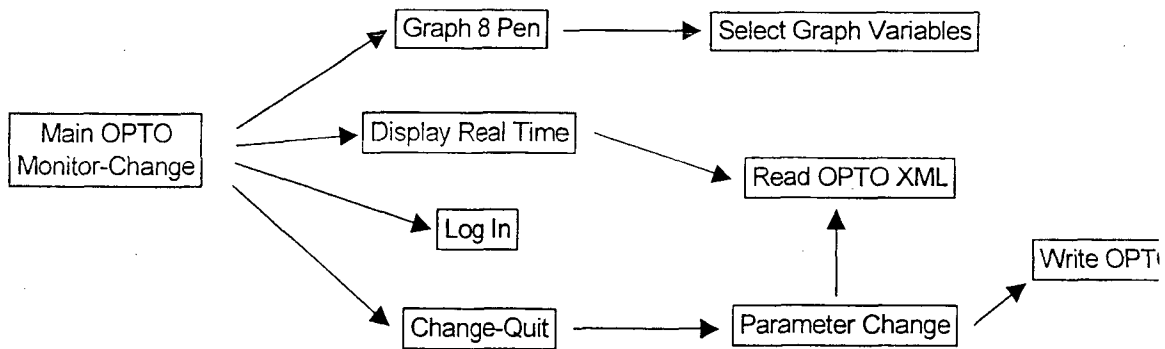Figure 2. Wiring panel that housed the PLC modules.

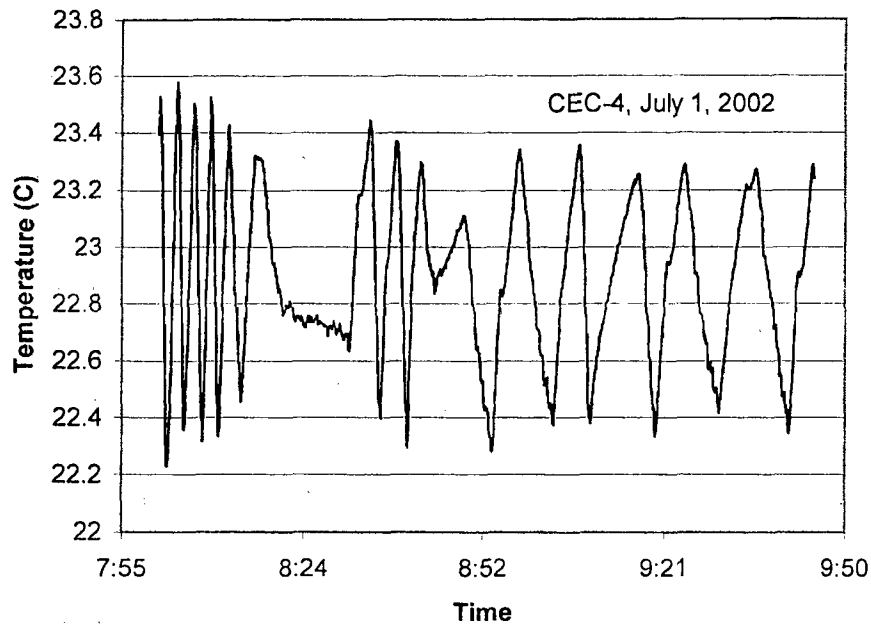Figure 3. Structure of the LabVIEW interface showing the sub-VI's.

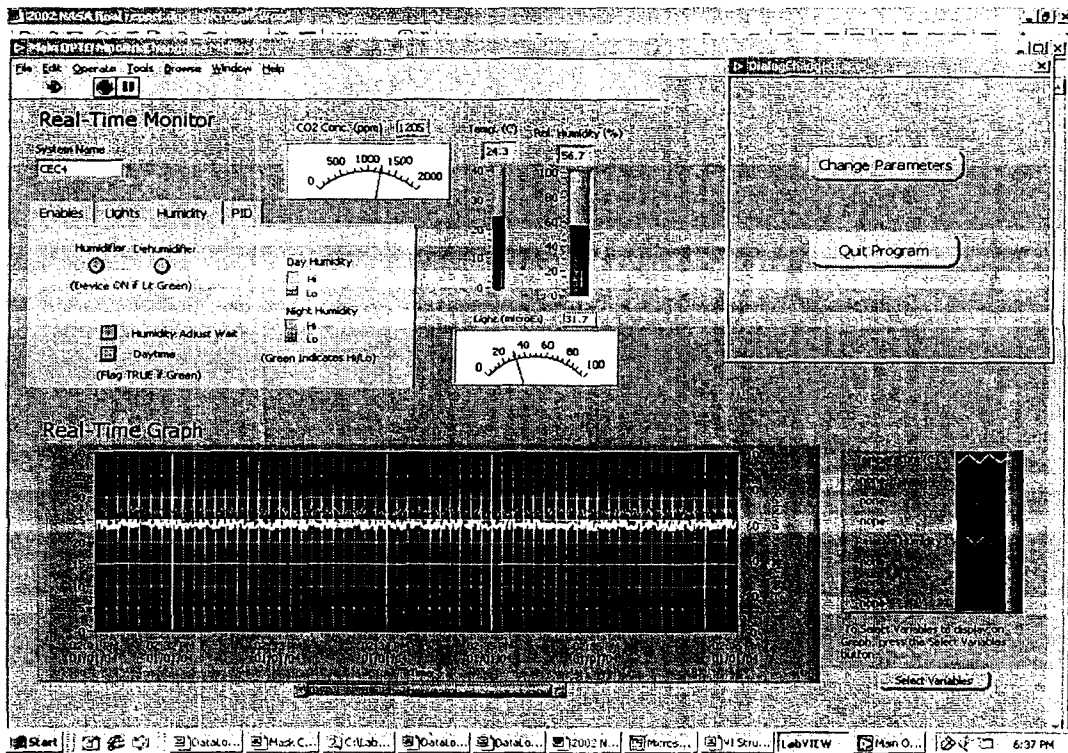**Figure 4. Typical temperature fluctuations in the growth chamber under autonomous control.**

Figure 5. Real time display using the LabVIEW software.