A Low Cost Robust Architecture with High Connectivity for the Control of refrigeration chambers

A. Marti Campoy, Jose Carlos Campelo, Juan Jose Serrano and Juan Vicente Capella
Department of Computer Engineering
Universidad Politecnica de Valencia
46022 Valencia
Spain
Email: amarti@disca.upv.es

Abstract— *Computer-based control of refrigeration chambers must be fault tolerant because of their critical mission. Poor or no response in case of wrong operation of the system may result in the loss of the products. Achieving fault tolerance involves all elements in the control system, including microcontrollers and the interconnection network. This work presents a practical architecture for distributed control of refrigeration chambers with fault tolerance and low cost as main characteristics. The proposed architecture has been implemented and tested in the laboratory using fault injection. The system has also been implemented in actual facilities for conservation and maturation of citric fruits, showing its high robustness.*

1. Introduction

Refrigeration chambers are widely used to conserve and, in some cases, to process fruits and other foodstuffs for market distribution. Not only temperature, but also other environmental parameters like humidity and CO2 levels must be accurately controlled in order to obtain high-quality products. Deviation from desired values may damage the stored goods, and even fruits may be lost if malfunction lasts long. Modern industrial control systems offer several advances in control techniques, allowing the development of very efficient algorithms that control the process in a quasi-optimum way. However, it is necessary to analyze and improve other aspects and features of control systems, like fault-tolerance. Fault-tolerant systems present improvements over conventional systems in terms of dependability attributes [1]: reliability, safety, security, servicing, availability and integrity. A system is fault-tolerant when it is able to continue working despite the occurrence of permanent, transient and intermittent errors. On the other hand, a system has a safe response to unrecoverable errors when it is able to stop at a known state, with no risk for the process that it controls. In this sense, the need to evaluate complex and critical systems, highly complex computational industrial systems or systems in which failure may result in high economical losses or even worst, loss of human lives, raises the issue of dependability. Fault tolerance techniques allow improving dependability factors in control systems, dealing with faults, errors and failures. Common techniques are fault prevention, fault tolerance, fault elimination and prediction [1]. The main objective of these techniques is evident: the reliable operation of the system. Advances in computer performance and low cost technology have enabled the development of fault tolerant control systems with a negligible price increase. In refrigeration chambers, the main goal regarding errors and failures is fault tolerance; that is, keeping system operation in occurrence of transient and permanent errors, to guarantee product integrity. For this end in this paper the use of a distributed control system is proposed. Distributing the functions of the system over several nodes allows the use of simpler, cheaper nodes, which results in more reliable nodes. In addition, the use of cheaper nodes implies that the use of replicates (hardware redundancy [2]) is an economically viable option and therefore, fault tolerance can be improved at low cost. However, the design of fault tolerant systems must take into account all the elements in the system: the microcontrollers used to control the system, the network used to interconnect the microcontrollers, and the sensors and actuators that measure and modify the physical parameters of the system. All elements in the system must include mechanisms to detect faulty operation and shut-down those components that present malfunction, transferring their functions to alternative devices. In order to reach the desired degree of fault tolerance at low cost, the following considerations have been assumed during system design:

- The control nodes, which monitor the environmental conditions of the refrigeration chamber and perform control over actuators, must include some fault tolerance, for self-disconnection in case of faulty operation.
- The control system must be distributed in order to be cost-efficient, flexible and fault tolerant by redundancy, since a control node may operate the refrigeration chamber of a failed node.
- The sensors and actuators, with their signal conditioning hardware, must be independent of the control node to continue normal operation if the control node fails.

- A reliable and secure field bus must be used to interconnect all the nodes.
- Communication between nodes must be guaranteed even if some segment of the field bus fails.

This work presents a comprehensive architecture for fault tolerance in refrigeration chambers, dealing with all the elements inside the facilities. The fault tolerance of the proposed architecture gets support from two techniques: hardware redundancy and checkpointing. Also, the system is able to include more complex methods to detect faults, like Fault Detection and Isolation (FDI) and Fault Detection and Diagnosis (FDD) for sensors fault [3] and for actuators fault [4]. These techniques can be used in refrigeration systems, like described in [5] [6]. However, the complexity of techniques described in these works may demand for high-performance, expensive microcontrollers. Since one the main goals of this work is to get fault tolerance with low cost, the use of low-cost, low-performance microcontroller precludes the use of the cited methods. Experimental results show that the proposed architecture, with its simpler fault tolerance mechanisms, is enough to guarantee the safe operation of refrigeration chambers.

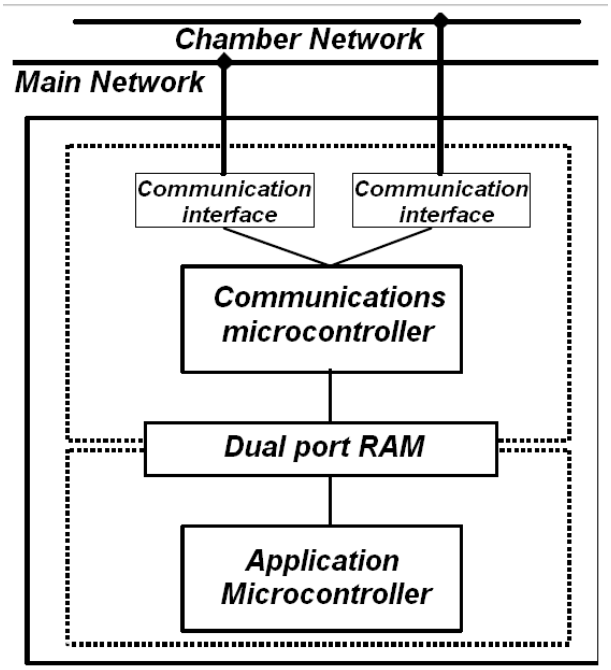

Fig. 1: Internal structure of control node

2. System architecture

Following, the components of a control system for citric refrigeration chambers are described. Although the system presented in this paper has been used in citric refrigeration chambers, the control system can be applied to other similar facilities aimed at conserving vegetables or growing-plant facilities like greenhouses.

2.1 Control node

The function of the control node is to run the algorithms that control the environmental parameters in the refrigeration chamber. It analyzes the data received from the sensors and sends the adequate commands to the actuators. The control node contains two microcontrollers, one for control purposes and another for communication purposes. The microcontroller used for the execution of the control algorithms is Silabs C8051F043 [7]. It has up to 25 MIPS throughput, 4352 bytes of internal RAM, 64 KB of Flash memory, two serial ports, SPI and I2C buses, 5 timers/counters and 1 watchdog timer. For the communication tasks and watchdog processor the Atmel CANary T89C51CC01 microcontroller was chosen [8]. It has up to 5 MIPS throughput, 1280 bytes of internal RAM, 32 KB of Flash memory, 2KB of on-chip EEPROM, built-in CAN controller, 34 I/O pins, five channel 16-bit PCA, serial ports, 3 timers/counters, 1 watchdog timer and 10-bit A/D converters. Both microcontrollers are interconnected through a dual-port memory. Microcontrollers exchange data from/to the network through this memory. These microcontrollers form the core of the fault tolerant node of the distributed control system. The communications microcontroller implements the local checkpoint of the two-level checkpoint system, storing the state of the control algorithm, which may be recovered by another control node if the Silabs microcontroller fails. Also, the communications microcontroller operates as a watchdog processor for the control microcontroller. The control microcontroller sends data about its execution flow ([9] [10]) to the communications microcontroller through the dual-port memory. The communications microcontroller verifies that the execution flow is right, resetting and even disconnecting the control microcontroller [11]. Finally, the communications microcontroller can exchange messages between the main network and the chamber network even if the control microcontroller fails, allowing other control nodes to operate the chamber with a faulty control microcontroller. See figure 1.

2.2 Sensors node

In order to facilitate the integration of different sensors another kind of node has been developed. This simple node collects data from several analog sensors (temperature, humidity, and gas for example) distributed inside the chamber and sends the data values to the CAN network. Any subsystem requiring those input data can read the messages from the network (the main node, display node, actuator node, or any additional node). This node has not a fault tolerant architecture to minimize system cost, but in order to improve system reliability it is possible to replicate the node and/or the sensors to have several measurements. For these functions a simple node architecture based on the

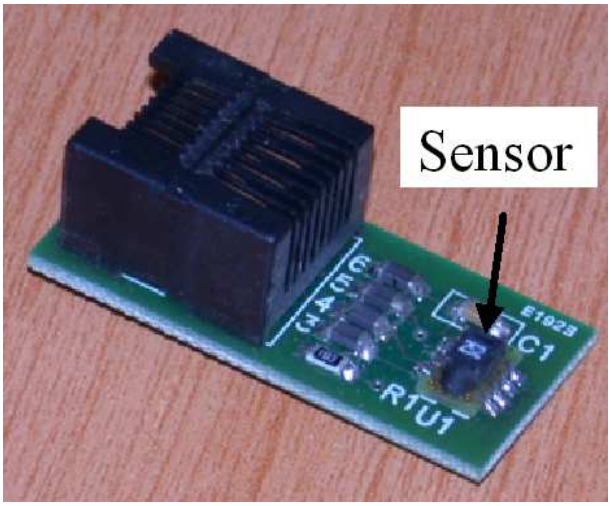

Fig. 2: Serial linked temperature and humidity sensor from Sensirion. Its low size, low cost, makes it suitable for replication with fault tolerant purposes

Fig. 3: Example of sensor node. This node supports up to six serial-linked sensors. These sensors may be identical for replication purpose, or may be used for different measurements

CANary T89C51CC01microcontroller has been used. The sensor node may present transducer-dependent circuitry to measure the physical magnitudes, so several versions of this node may be needed. However, the use of intelligent sensors with digital interface by means of serial links, like the temperature and humidity sensors developed by Sensirion [12] helps reduce the number of different versions of sensor node. See figure 2 and figure 3.

2.3 Actuators node

These nodes receive the commands from the control node and switch on/off their outputs which are connected to different subsystems (i.e. heating/cooling machines, ethylene injectors). The same CANary T89C51CC01 microcontroller is used to build this node. Like in the sensors node, there is no built-in fault-tolerance, but actuators nodes may be replicated to operate the same set of actuators

2.4 Personal computer

The personal computer performs two operations. First, the programming, tuning and supervision of all the nodes, allowing a human operator to control the system's operation. Second, it helps with the fault-tolerance assessment by periodically storing the state of each refrigeration chamber. If some control node fails, another node can take control recovering the state from the personal computer. This is the remote checkpoint of a two-level checkpoint [13]

2.5 Display node

The system offers the possibility to include display nodes: it is a node with a 4-line LCD display and a keypad. This node displays information about the process to control and allows the operator to set some control parameters without moving to the personal computer. As its task is not critical no fault tolerant solutions have been adopted. The core of this node is the same CANary T89C51CC01 microcontroller used in the sensor node.

2.6 The net

Work communication is structured over two CAN [14] networks in a hierarchical way: the main one interconnects the personal computer and control nodes of all refrigeration chambers. In the second level of the hierarchy, there is a network local to each chamber that connects all sensors and optional equipment to the chamber's control node. Since the control node belongs to two networks, the use of a communications microcontroller is necessary as described in the section above. The baud-rate chosen as a tradeoff between bus length, noise immunity and frequency of messages is 125 kbps. CAN network is a robust, fault-tolerant fieldbus that allows the nodes to monitor the correctness of communications.

2.7 Bridge node

This node forms part of the communications subsystem of the fault-tolerant-system, and performs both recovery and monitoring functions. If the communications microcontroller in the control node fails, all the nodes inside the refrigeration chambers continue working, but they are unreachable from any node able to run control algorithms. In this case, the bridge node starts working, building a bridge between the isolated sub-network and another sub-network, routing CAN messages from the sensor and actuator nodes and the control

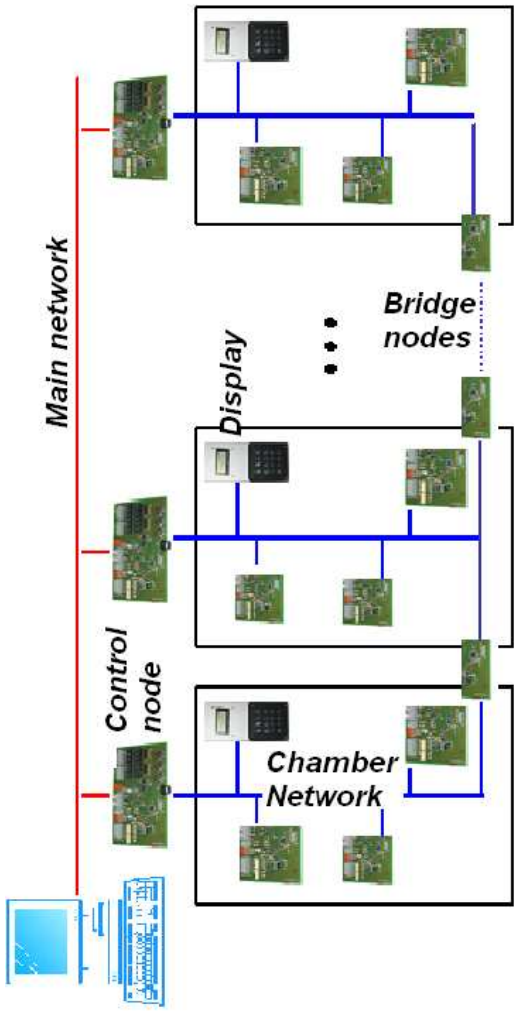

Fig. 4: Overall diagram of the control system architecture

node. Despite the high programming complexity of the system, the very low cost of the bridge node makes this approach a very interesting proposal. The bridge node is a Silabs C8051F043 with an external second CAN controller connected by the SPI bus. In this way, the interconnected subnetworks remain isolated until failure in one node forces their interconnection.

Figure 4 shows an overall scheme of the control system, with all kinds of nodes.

3. Fault tolerance procedures

This section presents the mechanisms used for system recovery after failure or damage. Fault tolerance is based on three actions: monitoring of all elements to detect malfunctions; alerting human supervisor of malfunctions; and, when possible, system recovery from failure. The way these operations are performed depends on the element that is under supervision. Sensors are supervised by the microcontroller of their own node. This supervision depends on the characteristics of each sensor (i.e., a high slope in the signal from a temperature sensor may indicate malfunction). If the control node receives a sensor-malfunction message, it alerts the human operator through the display nodes and the personal computer. The system may recover from this kind of error if the faulty element is replicated. Same procedures apply to actuators and their corresponding nodes. Although sensors or actuators operate normally, failure may occur in the sensor/actuator node itself (microcontroller or power supply failure). To allow the control node to detect this scenario, every sensor/actuator node sends a watchdog message periodically. If the control node stops receiving these messages, it alerts the human operator and uses data from a replicate node, if available. Failure in the microcontroller that executes the control algorithms is detected by the communications microcontroller with a watchdog operation. When the control microcontroller fails, the communications microcontroller broadcasts a message with the latest state of the control algorithm (first level checkpoint). Another control node, or even the personal computer, assumes the functions of the faulty microcontroller, and controls the refrigeration chamber. The system continues working without loss of quality, and the human operator is notified through the personal computer and display nodes. Failure in the communications microcontroller is detected because the control microcontroller stops receiving any data from the nodes inside the refrigeration chamber as well as from other control nodes or the personal computer. However, the control microcontroller is unable to send any message to alert the human operator and other nodes. Then, the bridge node in each refrigeration chamber detects any failure in the communications system because it does not receive any message from the control node. In this case, the bridge node sends a message in order to allow other control nodes in the system to take control of the refrigeration chamber. This message is similar to that sent by the control node when the control microcontroller fails, but in this case, the state of the algorithm is recovered from the personal computer (second level checkpoint). Finally, the bridge node builds the bridge allowing access to the nodes inside the refrigeration chamber, and the operation of the chamber continues with no loss of quality. If the entire control node fails, the system recovers as explained above: failure is detected and notified by the bridge node, and the control state is recovered from the personal computer. The personal computer is not monitored; as a consequence, failure of this element must be detected by the human operator. Since the main function of the personal computer is to interface with the human operator, it is not a critical element. Only when some control node fails and the system state must be

recovered from the personal computer, the system may lose quality if the personal computer is not working. If failure on the network makes some element unreachable, the system acts as if there were a failure in this element, and proceeds as above. If any element recovers from failure, the system behaves as if the element still were under failure mode. In this way, if the element presents a high number of transient failures, its behavior will not overload the system. Only the human operator can re-run an element that has stopped due to failure. Table 1 summarizes the procedures for failure detection and recovery.

Table 1: Summary of fault-tolerance techniques

Failure in	Detected by	Recover from
Control microcontroller in control node	Communications microcontroller in the same node	First level checkpoint. Another control node or the personal computer
Communications microcontroller in control node	Bridge node in the same chamber	Second level checkpoint. Another control node or the personal computer through bridge node
Control node	Bridge node in the same chamber	Second level checkpoint. Another control node or the personal computer through the bridge node
Sensors	Microcontroller in the sensors node	Sensor replication
Sensors node	Control node	Node replication
Actuators	Microcontroller in the actuators node	Actuator replication
Actuators node	Control node	Node replication
Display node	Without fault tolerance	
Bridge node	Without fault tolerance	
Personal Computer	Without fault tolerance	
Network	Can field-bus is used, with built-in fault-tolerance	If some node becomes unreachable the system acts in the same way as when the node fails

4. Experimental results

Two kinds of experiments were conducted. First, laboratory experiments were performed to force the control system and estimate the number of errors supported by the system. Second, one real facility was monitored for more than four months. In the laboratory experiment, the number of control nodes was nine, and there were between three and six nodes in every subnet (sensors node, actuators node and display node). The number of bridge nodes was four. Progressively, the different nodes were forced to fail, including failures in the control microcontroller and communications microcontroller of the control nodes. The system worked at full operation except in one case: when only the control microcontrollers were forced to fail, the main network became overloaded for a number of faulty microcontrollers of five. This is because all the messages from the out-of-control chamber are routed through the main network, that is, the traffic is five times the planned traffic. Five is not an absolute value. It depends on the number of sensors and actuators in the subnets; that is the main network may become saturated when only two or three control microcontrollers fail. To avoid this problem, if a control node operating a second refrigeration chamber detects collapse in the main network, it will activate the bridge node to create an alternative route to the sensors and actuators. In the case study, the system was implemented in a facility for orange maturation. This facility consists of three refrigeration chambers with two sensor nodes, one actuator node and one display node in each chamber. Also, two bridge nodes interconnect the three refrigeration chambers. In addition to the personal computer for monitoring purposes, a second personal computer was used to record network traffic and operating parameters in the refrigeration chambers. During four months failures were randomly forced in some devices and nodes, without notification to the human operator. After the four testing months, all the environmental parameters (temperature, humidityï£¡) of the three refrigeration chambers were within desired and safety values, and the quality of the oranges was excellent.

5. Conclusions

The control system described in this paper is a distributed system formed by a hierarchical network system with the necessary attributes of flexibility, scalability, dependability, easy installation and operation. Furthermore it provides a perfect adaptation to the necessities of this type of installations and facilities at low cost. This distributed system is constructed around two CAN networks. The nodes are implemented by means of low cost, 8-bit microcontrollers. In the nodes design we used a small number of different nodes, each one for a generic task, with the goal in mind to reach cost-effective production. Different fault-tolerant techniques were used in the system: the control node includes two watchdogs -one watchdog timer inside the microcontroller and a watchdog processor implemented in the second microcontroller- to detect wrong operation in the main microcontroller. A two-level checkpoint technique is used to recover system operation if a control node fails. The communications are independent of the control microcontroller, allowing alternative nodes to take control of any refrigeration chamber. The bridge nodes permit the development of a network on the fly, which allows the control system to continue operating even if the communications microcontroller fails. This control system has been implemented in different actual facilities. The application of the control system in agricultural facilities has allowed us to verify the correct operation, installation and maintenance of the system.

6. Acknowledgments

This work has been suported by MEC under project

References

[1] Guide de la surete de fonctionnement. Laboratoire d'Ingenierie de la Surete de fonctionnement (LIS). 1996. Cepadue-editions.

[2] W. Dunn. Practical Design of Safety-Critical Computer Systems. July 2002. Ed. Reliability Press

[3] R. Xu and C. Kwan. Robust Isolation of Sensor Failures. Asian Journal of Control, Volume: 5, Issue Number: 1, Page(s): 12-23

[4] Kwan, C. Xu, R. A note on simultaneous isolation of sensor and actuator faults. IEEE Transactions on Control Systems Technology. Volume: 12, Issue: 1 On page(s): 183- 192

[5] Shengwei Wang, Jianying Qin. Sensor fault detection and validation of VAV terminals in air conditioning systems. Energy Conversion and Management. Volume 46, Issues 15-16, September 2005, Pages 2482-2500

[6] Thybo Claus, Izadi-Zamanabadi Roozbeh. Development of fault detection and diagnosis schemes for industrial refrigeration systems - lessons learned Proceedings of 2004 IEEE Conference on Control Applications. September 2004. pp. 1248-1253.

[7] Silicon Laboratories. C8051F040/1/2/3 MCU Family. www.silabs.com

[8] Atmel Microcontrollers. Atmel T89C51CC01. www.atmel.com

[9] Yung-Yuan Chen.Concurrent detection of control flow errors by hybrid signature monitoring Computers, IEEE Transactions on Volume 54, Issue 10, Oct. 2005 Page(s):1298 - 1313

[10] M.S. Hefny, H.H. Amer. Design of an improved watchdog circuit for microcontroller-based systems Microelectronics, 1999. ICM '99. The Eleventh International Conference on 22-24 Nov. 1999 Page(s):165 - 168

[11] A. Mahmood, E.J. McCluskey. Concurrent Error Detection Using Watchdog Processors-A Survey IEEE Transactions on computers. February 1988 (Vol. 37, No. 2) pp. 160-174

[12] Sensirion. The sensor company. www.sensirion.com

[13] Rubio, A, Ors, R. 2003. A Comparative Analysis of the Reliability of Simple and Two-Level Checkpointing Techniques in two different Distributed Industrial Control System Architectures. Systems Analysis Modelling Simulation. Vol. 43, No. 7, pp. 945-957.

[14] CAN 1991. Specification Version 2.0. Robert Bosch GmbH.