To Alex with bes wishes Nick

A HIERARCHICAL MULTICOMPUTER SYSTEM FOR MEDICAL CARE

by N. DeClaris

Maryland Institute for Emergency Medical Services Systems
and College of Engineering
University of Maryland

ABSTRACT

This paper reviews the medical environment which gave impetus to computer-based automation in critical and intensive care units. It formulates automation functions based on the clinical experience of the last decade and on the expected infusion of "intellegent" instruments in the hospital. Three levels of a dedicated hierarchical computer-network are then identified which afford hardware-software trade-offs. Horizontal partitions of each level lead to designs of universal modules employing 8-bit, 16-bit and very likely 32-bit microprocessors.

INTRODUCTION

Technology is bringing about significant improvements in the quality of medical care in this country. In the area of critical care medicine the technology impact is characterized by two trends: 1) ever increasing number of intelligent instruments (microprocessor-based diagnostic and therapeutic aids) brought into the hospital and 2) medical profession acceptance of holistic approaches (systems methodology) for the management of the patients. This paper discusses a novel microprocessor-based computer network dedicated to automated patient care which takes into account both of these trends so as to avoid near-future obsolescence.

It begins with a review of the medical environment in which computer-based automation evolved in the last decade and a brief look at the recent trend toward the use of computer-networks for this purpose. It is concluded that at this time a formulation of all the currently known automation requirements needs to be made. Then using them as a set it is possible to configure, on the basis of hardware-software trade-offs as a design criterion, a network architecture of dedicated computer-based modules that will serve well the clinical environment of this decade. The key idea here is to start with the overall system specification and then through successive decompositions that afford convenient hardware-software trade-offs to come up with primitive universal modules that will constitute the building-blocks of the complete system.

THE MEDICAL ENVIRONMENT

Trauma Center, a concept which requires "men" and "machines" working smoothly together for the medical management of critically ill patients, proved to be very effective within a few years of its conception by saving lives that otherwise they would have been lost (Cowley, 1977). A key feature in the management of critically ill patients is that it requires painful attention to detailed and repetitive routine tasks which nevertheless must be executed accurately. As these tasks became standardized, it also became clear that, computer-based automation could take over the execution of a number of these routine tasks thereby enhancing considerably the life-saving potential of the trauma center and lowering the cost per patient of the intensive-care they provide (DeClaris and Cowley, 1978). It was also early recognized that the introduction of computers in the management of critically-ill offered the possibility of improving the quality of medical care in the trauma center through data-bank acquisition and protocol development (DeClaris, Cowley and Trump, 1979). That is, the use of the computer in the trauma center setting while enhancing the management of the current patients in the ward but it should also ensure that the experience acquired through this management would improve still further the quality of medicare-care future patients will recieve. The routine functions that have been carried out traditionally during the decade of the 70's in the hospital via computer-based automation are:

1) to observe and record periodically the physiological status of the patients (patient monitoring),
2) to determine physiological quantities that can not be measured directly (e.g. cardiac output),
3) to display on command waveforms and graphs or tables on the past history of the patient (charts),
4) to carry out numerical calculations of dosages needed for patient care (e.g. fluids, drugs), and in some cases
5) to deliver (or monitor the delivery) of prescribed dosages of therapeutic agents.

The above functions make rather small demands upon commercially available computer hardware. In the decade of the 70's the use of the computer in the hospital could only be economically justified if it could take care simultaneously a sufficient large number of patients in a time-sharing mode (6 to 12 patients). Thus the natural configuration for such an automated patient monitor scheme consisted of a central minicomputer connected:

1) to a set of physiological signal monitors (typically EKG, Temperature, Blood Pressure) and one CRT display at the bedside of each patient and,

2) to a remote terminal for accessing the computer at the central "nurses" station.

By the middle of the 70's remote terminals were added at the bedside of each patient so that the computer could be accessed from these terminals as well. About the same time, in some places, terminals were also added in the clinical chemical laboratories from where data on blood, plasma and urine analysis may be entered to the computer for all patients. These remote terminals had of course no stored-program capability and the burden of information storage and processing fell solely upon the central minicomputer, which in the hospital setting meant that it must be backed-up for reliability purposes by a second stand-by minicomputer. The overall situation in most cases was less than satisfactory and voices among the medical profession began to be raised as to the advisability of using expensive machines (both to buy and to maintain) for paramedical aid purposes (record keeping and charting) when in fact major improvements in the quality of medical care are expected to come from the direction of the use of computers to aid directly the physicians and the patients (long-range therapeutic plan formulation, protocol developments, etc.). As early as 1977 it became evident that automated intensive care system in the hospital will have to rapidly evolve to be truly computer-network structures (DeClaris, 1978). Fortunately at about that time LSI technology lowered considerably the cost of individual electronic components and it brought about what has come to be known as the "microprocessor" revolution. The advent of inexpensive "microprocessors" and "chip-memories" had a major impact in the hospital by making available at the bedside of each patient "intelligent" computer terminals and "intelligent" instruments; that is hardware with memories and stored-program capabilities. However the communication requirements between diverse "intelligent" machines are not easily satisfied in general, in fact, in the intensive care environment, the solution to this problem influences in sifnificant way (conceptually as well as in cost) the automation processes. There is a tendency in man-made systems to grow in a given direction even after the evolutionary forces which gave it impetus no longer exist. Thus, one approach that has been taken in some cases is to modify the single computer structure used in the intensive-care unit of the hospital during the early 70's so as to become a multicomputer "star-configuration". The "star-configuration" consists of a central minicomputer (possibly backed-up by another identical one for reliability purposes) wired directly (through serial-ports) to several computer-based terminals and instruments. However while the star-configuration can be readily implemented with commercially available components so as to function in the traditional medical environments, its well-known weaknesses of poor-growth capability and poor fault-tolerant operation makes it a rather unattractive (from the cost as well as the reliability point of view) computer-architecture for meeting the future automation aspirations of the continuously improving medical environment of critical care. In the next section we briefly discuss additional automation functions that are most likely to be realized in computer-based medical care and we conclude with the general description of a hierarchical multicomputer structure especially designed for this purpose.

EVOLUTION OF THE STRUCTURE

From the discussion in the previous section it is clear that automation of the medical-care is headed toward a multicomputer structure. Such a structure must provide communication between diverse intelligent instruments and terminals. Moreover it must make possible another modality of medical care: closed-loop configurations that can initiate and automatically administer therapeutic agents to the patients via software that include pharmacodynimic considerations as well as model validation algorithms (DeClaris and Kempner, 1978). Thus to the five automation functions that emerged from the clinical experience of the last decade, as listed in the previous section, we must now add five additional functions that must be accommodated by the multicomputer structure. These are:

6) Multi-instrument communication

7) Closed-loop control

8) Failure-tolerant operation

9) Mangement of distributed data-base

10) Learning and decision making capability.

It is possible to elaborate in much greater detail these functions and thereby to come up with a set of overall system specifications. The next step represents a giant system design challenge. How does one choose a connection of computers and a set of operating systems so as to meet information and control requirements in an economically acceptable way? Design methodology for this purpose does not yet exist. The author has faced this problem in the medical care environment and has come up with an approach which provides design guidelines. The essense of the approach (to be described elsewhere) is for the network configuration to evolve gradually as a result of successive design stages. At each stage important algorithmic realization decisions are made on the basis of hardware-software trade-offs. The identification of the design stages is through groupings of the system requirements into subsets that constitute realizable computer subsystems "partitions" and subject to acceptable connectivity (hardware-software) constraints implied by a chosen "decomposition". In this case the decomposition of the system requirements begins with a "vertical partitions" that reveal computer CPU specifications. Thus starting with the bedside instrument as the

0-level, the computer requirements are grouped into three vertical levels as shown in Table I.

Table I. Vertical Partitioning of the Hierarchical Computer Network

Level 0	Bedside environment; including monitors, instruments, actuators (microprocessor-based or not) Displays, Printers etc.
Level I	Microcomputers carrying well defined tasks with quick response to environment; single-unit failure causes no loss of information and control functions.
Level II	Microcomputer supervised communication and control between units; provides guidance and previously developed programs: single-unit failures are diagnosed and short-term tolerated while essential functions of the system go uninterrupted.
Level III	Microcomputers (or minicomputers) for large data base processes including program and protocol development; auto-diagnatics; machine servicing without shotdown.

The computer levels implied by the vertical partition are hierarchical in the sense that:

1. the size of the memory and the complexity of the algorithms implemented increases with increasing level number
2. fault-tolerant capability increases with increasing level number
3. the demand for real-time operations of repetitive routine tasks increases with decreasing level number.

The next step in the evolution of the structure is to identify "horizontal partitions" are each level. At Level I, the system requirements can be put into disjoint sets which have no simple data relationship. Thus horizontal partitioning at Level I is based on the separation of sensor-related and operator-related groups of processes to be carried out concurrently by a parallel connection of (loosely coupled) units. Further design considerations lead to the development of a universal bedside microprocessor-based module (BeMi) which constitutes the building block for the first level of the envisioned hierarchical computer structure. The Z-80 microprocessor (8-bit architecture) proved to be a good choice for the CPU at this level (DeClaris and Paratore, 1980).

The horizontal partitioning of Level II is determined by the strong connectivity requirements (communication and control) that must be incorporated into the system. The result is another universal module at Level II which consists of a CPU with time-sharing capability and an intelligent communication port. Preliminary design seems to confirm that a 16-bit microprocessor (the Z8000) serves well as the main CPU of the module (together with a Z-80 for the communication port) and it can accommodate 4 to 6 BeMi's in a double buss configuration.

At this stage of the design no horizontal partitioning of Level III is contemplated- whether such partitions are indeed needed will depend on the characteristics of the latest generation of 32-bit (or higher) microprocessors soon to become widely available commercially.

CONCLUDING REMARKS

The three-level structure of the hierarchical system outlined above was conceived several years ago at a time well ahead of any announcement about the commerical availability of the Z8000 and the 32-bit microprocessor. In a way, the fact that these microprocessors became available in time to facilitate the design of the structure is a testimonial to the anticipation foresight that was built into it. I am greatly indebted to Dr. R.A. Cowley, the director of MIEMSS, for providing the environment to persue this approach and to my student Bernardo Paratore for getting heavily involved in the design of Level II.

REFERENCES

Cowley, R.A. (1977). Trauma Center: A New Concept for Delivery of Critical Care. J. Med. Soc. N.J. Vol. 74.

DeClaris, N., R.A. Cowley and B.F. Trump (1979). A Systems Approach to Intensive Care Medicine. The American Surgeon, Vol. 45.

DeClaris, N. and R.A. Cowley (1978). Automation Principles for the Management of the Critically Ill. Proc. Allerton Conference on Communication, on Communication, Control and Computing, U. of Illinois.

DeClaris, N. and K.M. Kempner (1980). Control and Computing for Automated Intensive Care of the Cardiovascular System. Proc. Allerton Conference on Communication, Control and Computing, U. of Illinois.

DeClaris, N. (April 1978). Medical and Engineering Principles of Automated Intensive Care System. Invited Lecture at the BIOSIGMA Conference in Paris, France.

DeClaris, N. (to be published). A State-Space Framework for Designing Multiprocessor Networks with Applications.

DeClaris N. and B. Paratore (1981). BeMi- A Universal Bedside Microprocessor-Based Module for Patient Care. Bioengineering, Proceedings of the Ninth Northeastern Conference W. Welkowitz ed., Pergamon Press.