MULTIMICROCOMPUTER SYSTEM FOR BUILDING WITH FULL SERVICE FACILITY AUTOMATION

J. F. TIRADO , J. J. RUZ , M.MELLADO

Dept°Informatica y Automatica, Fac. fisicas
Univ. Complutense, Madrid-3. Spain

ABSTRACT

The system here presented has been designed to be used on lodging establishments or institutions such as hotels, apartment buildings, hospitals, etc. The system picks up building generated data in order to provide the actual status of the building at any moment and performs status-dependent operations to provide: a) service to guest; b) security in the instalment, and c) statistical and accounting information for management purposes.

The aforementioned operations are realized by the system within two distinct forms:

1) An automatic form which takes care of control of power consumption, accounting on service in each room (sell points, taxes for special services, fire-prevention, automatic wake-up, service) updating and reporting of statistical data for management.

2) An interactive form which through CRT stations attached to the system performs customer check in/check out; invoicing, query operations on guests, service lock/unlock, building status report, ...

The system has been implemented as a microcomputer network made up by a microcomputer in each room and a master unit of two tightly-coupled CPU's with shared-memory which is in charge of network controlling, terminal controlling, and processing functions.

1. INTRODUCTION

The present economic circumstances compel to introduce new technologies for management in this sort of business. The main purpose is to give the guests a continuous -24 hours a day- good service. Because of the increasing wages of employees, automation is the only way to achieve this result.

The automatic management provides some additional advantages: better and quicker information about building status and productivity, improvement of service to guests, energy saving, and greater security guarantees for the building.

At the time being, there exist several systems designed to handle these tasks. Usually, they cover back-office and front-office services but are not concern with automatic local control of rooms. (1,2)

A precedent of the system presented here is the KEY SYSTEM. It was developed by a personal team presently at INFORMHOTEL, and today it is installed in several hotels. The KEY SYSTEM gives back-office and front-office service as well as invoicing through terminals and automatic room status and control. This system is minicomputer-based and uses a disk-storage to hold all the building information.

The system described here evolves from the KEY SYSTEM looking for a low cost system in order to cover a market where the purchase of the KEY is too expensive. The system performs automatic room status and control and a number of useful front-office functions for the building management.

The low cost feature of the system leads to use the system main memory to store the building information, so avoiding the need for a disk storage. The memory CMOS technology is power fail protected. Back-up on cassette is included to realize maintenance functions. The system instalment cost is very low too, since no specific wiring is needed. The TV wire is used to communicate the different components of the system.

2. SYSTEM ORGANIZATION

The system is organized as a distributed network made up of a number of µP to perform the various tasks. There are two main groups of functions in the application: a) data collecting and room local operation functions; b) process and interactive functions. Each group is realized by a µP-based different device: the Room Terminal (RTC) and the host system.

There is a RTC in every room, and all of them are connected to the host by a single wire (multi-point configuration)(3) instead of individual ones, so reducing the data transmission wiring cost (Fig. 1). The RTC's are programmed to communicate with a host controller using a digital serial data communication protocol. In this multipoint network system communication between processors is half duplex so that only one processor transmits data at a given time. The communication software sees the host as a master and the RTC's as slaves. The master starts the transmissions sending commands to specific RTC on the link. The addressed slave receives and decodes a command, executes it and sends the resulting data back to the host.

2.1 Host System

At the time of designing the host system, it had to be considered the great difference of requirements of the building to which the system is going to be installed. Let think for instance, of the difference existing between a small hotel with 40 rooms, and only one point of quest attendance for reception, porter's desk, phone exchange, etc. served by one or two employees, and a luxury grand hotel with a thousand or more rooms, where the guest service is distributed at the several points, served by a large number of employees. Hence it was decided that the host system could take two basic configurations in order to cover the wide range of buildings where the system could be installed.

There are two main differences between a small building and a big one:

- The number of rooms to be controlled.
- The number of attendance terminal, from a basic configuration of one CRT and printer up to a maximum of 20 terminal. As the number of room and terminal increase for a given host system, the room updating times and the answering times on terminal to non-interactive operations can become incompatible with a good service which is the system main goal.

The two host system basic configurations are: one monoprocessor and one biprocessor where the tasks are distributed between two CPU's. The CPU used in both configurations is realized with μP 6800/

The tasks performed by the host system are the following:

- Supervision and control of all communications with the room RTC's
- Taking care of CRT and printer terminal
- Execution of automatic jobs started up by the real-time clock.

Therefore, the system throughput can be increased if two CPU are used. Simply, they distribute the tasks between the two subsystems.

Usually, distribution of tasks requires extensive subsystem communication, which in turn means a significant increase in hardware and software complexity. Subsystems based on 6800/02 microprocessors, however, can operate simultaneously on a common set of data with a minimum of interface hardware and virtually no software interaction. Each subsystem contains its own CPU with ROM and I/O interface. The RAM memory system, however, is shared and data are passed between subsystem across the shared RAM.

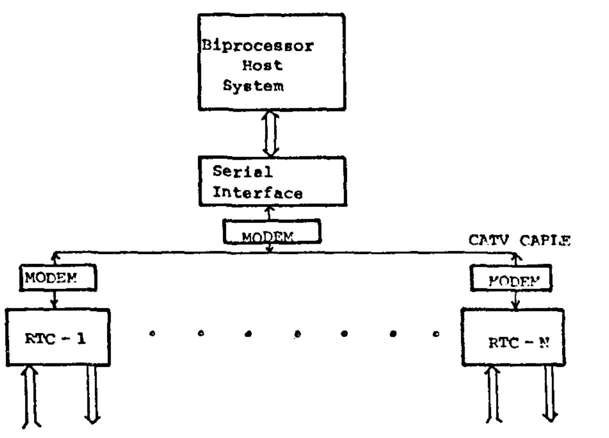

Figure 1

The use of shared RAM on thightly coupled multiprocessor architectures is popular. This particular design does not require implementing arbiters to resolve access conflicts to the shared memory. This important simplification depends on the phase relationship between two phase clock (Φ_1, Φ_2) of the μP. If the cycles are forced to maintain a constant 180° phase relationship, no RAM bus contention can occur. (Fig. 2)

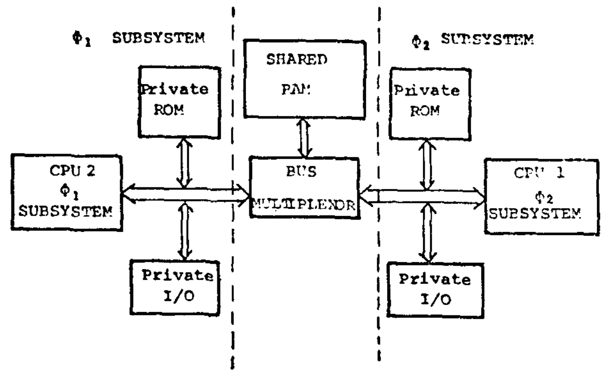

Figure 2

This host system disposition has been considered the most suitable for the application since it allows an increasing capability according to the growing demands in the future. The host system enlargement could be achieved through inclusion of new CPU's and more shared RAM (Fig.3)

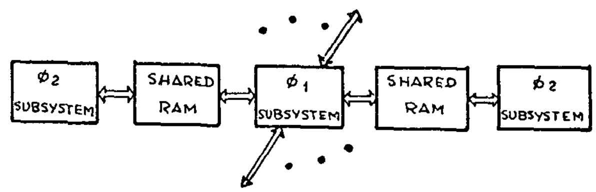

Figure 3

A growing number of new functions can be later commended to the system such as out-room sell-points control (restaurant, snack, laundry, shops, etc.) reservation and management information for hotel chains. In addition, since the I/O is memory, mapped part of the I/O equipment could be shared between various CPU's.

At the present, the yet built-systems are based on the monoprocessor version , and since the 64K available addressing is not enough, it has been necessary to expand the basic system addressing range by a bank switching method through four extra address lines. The system addressing capability is now 1Mb sufficient to meet the requirements of the applications. The bank switching has been chosen since it is an excellent and economical method for expanding the address range of a dedicated μP system (4).

2.2 Room Terminal

RTC main functions are data collecting and actuation on some elements in the room. The status of the elements are continously monitorized by the room terminal; the status includes the cosumption of products (food and drink) in the room refrigerator, blocking of up to 4 services. Where the assignation of a blocking to a service is given by the system's users, but blocking 3 is specially assigned to room access, accounting at fixed interval of special services sold to the guest, trace housekeeping activities, waitress call, "do not disturb" lock, waitress into the room, fire and burglar alarms. All this information is sent to the host system when they are asked for a specific command.

Furthermore, the RTC performs several actuations over room elements activated by the host system through the pertinent commands, they are:

- Defrost room refrigerator
- Turn blocking on and off, and change the cleaness status.

- Enabling of burglar alarms.
- Message notice to the guest in the room
- Automatic wake up
- Notice to waitress on a room
- Cut down the power consumption to a minimum when the guest leaves the room.

RTC is implemented with a microcomputer (COP-404) which performs all of those functions and supports, in addition, an auto-test program in charge of maintenance test operation in the room equipment. The number of commands acknowledged and executed by the RTC is 24.

All RTC's monitor the serial link all the time since any unit can be addressed at any time. The RTC can detect input stream and command sequence error. An input stream is checked by means of horizontal parity and a longitudinal redundancy check (LRC) character allowing detection of 1,2 or 3 bit errors in a message block. The host system performs a time check to detect burst errors in addition and after a number of succesive failed retries, the host puts out of service the concerned room, sending information to the system operators to carry out maintenance.

3. SYSTEM SOFTWARE

Two factors influence the software design in real time system: The maximum allowed response time to real-time events, and also the number of such events can be effectively handled at peak-load. Both factors are well defined for this application (5,6). The choice has been to design a very simple and compact real-time executive RTI with the following facilities: multitasking, event-driven processing and priority scheduling.

Multitasking is possible because task operation frequently involves more than CPU usage. A real-time task initializes a process and then waits for the process to complete will not use CPU time while it is waiting. Therefore while one task waits for an event (I/O operation) to complete, the real-time "executive" gives control of CPU to another task. For the RTI the tasks can be at one out of 4 states: dormant, ready, wait, and run. The RTI starts a task at occurrence of a real-time clock interrupt, a key-in -interrupt due to a key operation on a CRT station. If an interrupt requests service, while a started task is being executed, the task is suspended, and RTI determines the source of the interrupt. Then a program (interrupt driver) that has been specified to service the interrupt is executed. A task which was blocked from execution might now be unblocked. After every interrupt RTI re-evaluates which task should take control of the microprocessor.

System resources are granted to active task on the basis of priority and resource availability. When a task becomes blocked (wait for an I/O complete) RTI looks for another task to use the CPU; the chosen task is the one that has the highest priority and has access to all the resources it requires.

RTI is hence composed of the following functional program modules: interrupt analysis module; task control module; input/output control module and times control module. From the basis structure of the RTI the application software is divided into a variety of different tasks (Fig.4)

1: Recognition and execution of commands given by the system operator.
2: Writing of periodic reports such a Night-Auditor and Statistical Report for Management.
3: Communication with RTC's room-located.
4: Automatic miscellaneous functions (wake-up, cleaness status, automatic defrost). A task a function.
5: Location and sending a waitress to a room with information for operators through message on printer.
6: Maintenance allowing testing and on-line modification of parameters.

Every room has a portion of 36 bytes reserved in RAM memory. Thus, the memory size-needed for a 300-room hotel a typical configuration would be about 11K a satisfactory value. If the user wants to store additional data about the room, the reserved memory zone would have to be increased.

All the tasks handle a reserved memory zone, and transmit information through it. For instance, an operation command modifies that zone according to the order that is going to be sent to a given room, and the "get status and send order" task will effectively send it to the appropriate room: this way the reserved RAM zone is used as a link mechanism among tasks, whatever the configuration may be mono or biprocessor. In biprocessor configuration the reserved RAM is located in the shared memory. (Fig. 5)

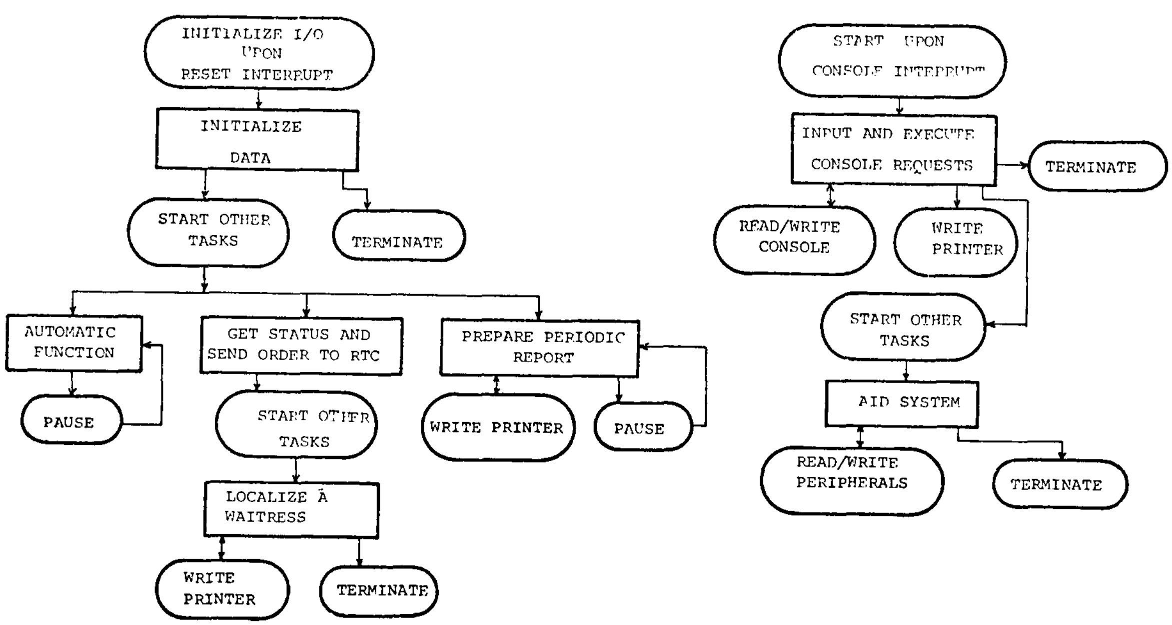

Figure 4

3.1 Get status and send order.

This program is ever communicating to the RTC's in order to get and update knowledge of the status information about every room, so the rest of programs could use it. Furthermore it sends to the rooms the pending commands. These commands can be transmitted; several in a stream forming an unique message.

The program also tests the message correctness putting RTC out of service if reiterated errors are detected at the answers or if the answer is not received after a predetermined interval (timeout error); these circumstances are presented both to the rest of the programs and the system operator through the reserved zone of memory dedicated to the room affected.

If a waitress call is performed at any room, the program starts a new task to localize a waitress in the nearest room to the calling room; then gives a notice of "call in suspend" and pass information to receptionist either if a waitress is available or not.

3.2 Automatic Functions.

What follows is a brief description of the main automatic function performed by the system.

3.2.1 Wake up: it works at 10 minutes intervals, triggering the alarm-clock buzzer in these rooms whose awakening requested time matches the current time. After 1 min. is elapsed, it turns off the buzzers in those rooms whose guests have not turned them off themselves. In addition, the wake-up function provides a print-up of the rooms whose users are awake or are not awake.

3.2.2 Automatic defrost and cleaness status: The automatic defrost is implemented to avoid the growing of a great quantity of ice on room's refrigerator. Every user definible number of days, and for a user definible interval time, the freezing equipment is tuned off and then tuned on again.

The housekeeping status is daily reset to a dirty state when the housekeeing tasks begin. After the waitress has cleaned the room, the housekeeping status is set to a "clean but inspection pending" state through a local action When the clean-inspector checks the room, he changes locally the status to a "clean and inspected" state and from now on the system considers the room as saleable.

3.3 Periodic Reports

There is two main periodic reports: A Night Auditor and a Statistical Report. The Night Auditor is reporting every day in the night at a user-definible time (3.00 a.m. typically). The program produces a print-out showing the room number; number of persons in the room; room prize, consumer of drinks and foods in the minibar in product units, and the guest name for every room occupied at the time; so this listing provides a complete vision of daily hotel business movements.

The Statistical Report is daily run along with the Night Auditor. It is a building production statistical summary for the current day and for the current month up to the running day. The reports inform about rooms rented along the day and occupation percentage, occupation average in the month, number of guests, rooms production, total and for each product(tonic,orange..) minibar production.

This information provides interesting data for management purposes enabling a better administration.

3.4 Aid System

This task is started by a command issued from the system main CRT (unique in minimal configurations). The AID to be executed requires special code without echo to be typed, in order to avoid unfairly handling of non-authorized operators. Once the AID is running, the main CRT only accepts special AID commands, different from the commands of the rest of the application.

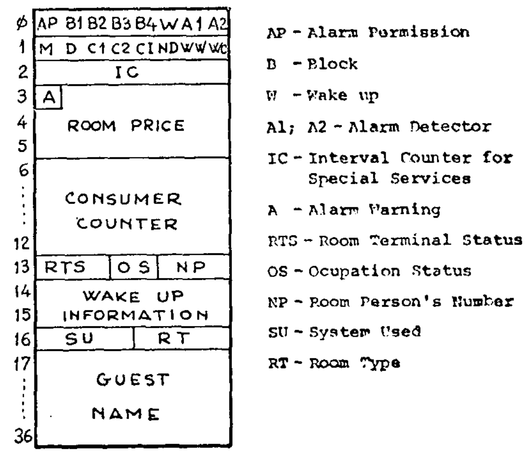

Figure 5

A menu of commands is presented to put the clock on time, to modify system parameters, modification and display in-binay, hex and ASCII of the room assignated words in memory, back-up and retraining of the system RAM memory through a digital cassette, execution of test function on the transmission line, RTC's testing. This program has proved to be of great worth at the installation and later maintenance of the system.

3.5 Input and Execute Console Requests

The operators interact to the system through terminal desk located at suitable points in the building. Every desk consists of a CRT station and a printer device; the printer can be shared for several CRT station or exclusive to a CRT. However, there can be desk without printer, but in any case at least one desk in the whole establishment must have one (it is recognized as the main desk).

The operator selects at the keyboard one command from the command menu table. Then, an interactive dialog is held driving to perform an access function particular to that command; guest check in, check out, etc. The dialog goes on progress as the operator answers the questions displayed by the system. The answers suffer an exhaustive syntatic analysis and they are checked to be consistent with the establishment particular parameters. If an error is detected, the system produces a warning message and asks the same question again. Moreover an answer dedicated area eases operator relies. This area is bound over the screen, so the display-answer cannot go beyond its limits. Before the operator reply is completed through pushing an "end-of-answer" keystroke, the operator is allowed to freely move the cursor along the bounded area to correct the answer.

Every desk is allowed to use a subset of executable command assigned at the initial configuration stage of the system. Thus, not all information may be accessed from a given desk.

4. CONCLUSION AND FUTURE DEVELOPMENT

The system described in this paper, a multimicrocomputer for building automation, is only the first step in a system's family for this application. The host system has been designed to be expansible, so new services could be incorporated in the future. Among these new services we could mention reservation management, phone billing, local -global billing, and local stock of another sell points inside the building. These can include restaurant, laundry, shopping center, snacks, etc. The system will be provided later on with a mass memory device to support less frequently used data about guests and rooms.

The production of this system is carried out by the Company INFORMHOTEL S.A. in Madrid (Spain). At the present the system is being installed in two 300-room hotels in Madrid and Barcelona.

COMMAND MENU: Table 1

Bi - Locking Services 1,2,3,4
Di - Unlocking Services 1,2,3,4
DA - Permission Burglar Alarms
AL - Turn on Room Alarm
DE - Turn on and off Wake up
ME - Turn on and off Message Warning
SC - Guest Check out
EC - Guest Check in
LA - Room's List with Alarm and Waitress Call
LV - Saleable Room's List
LB - Room's List with Lock Services
RA - Status Room's List
FH - Bill a Room
LC - Search a Guest by Name
LH - Display of all Room Status Including Guest Name and Products Consumer.

5. REFERENCES

(1) "Electronic Hotel Systems" Service World International pp. 34-80, March-April 1977
(2) P. Charlot, "Gestion Integrée des hotels par ordinateur" Informatique, n°120, 118-124, May 1978.
(3) C. Weitzman, "Distributed Micro/Minicomputer Systems" Prentice Hall, New Jersey, 1980.
(4) I. Lemair, "Indexed Mapping Entends Microprocessor Addressing Range" Computer Design, V-19, n°8, 111-118. 1980.
(5) D.L.Ripps, "On Operating Systems" Industrial Programming Inc. Jericho New York 1980.
(6) K. Burgeit and E.F. O'Neil, "An Integral Real-Time Executive for Microcomputers" Computer Design, v. 16, n°7, pp. 77-82, 1977.
(7) Microcomputer Components. Motorola Semiconductors Inc. 1979.