*MICROSYSTEMS: Architecture, Integration and Use*
*C.J. van Spronsen and L. Richter (eds.)*
*North-Holland Publishing Company*


# A MICROPROCESSOR-BASED HARDWARE MONITOR FOR PERFORMANCE EVALUATION OF COMPUTER SYSTEMS

Daniel Aljadeff — Ministry of Defense, Israel

Israel Koren — Department of Electrical Engineering, Technion - Israel Institute of Technology, Haifa 32000, Israel

The performance of operational computer systems can be evaluated using hardware or software monitors. Compared to software monitors, hardware monitors are portable, have a high bandwidth and do not interfere with the operation of the monitored system.
The design of a dual-microprocessor hardware monitor is presented. The first microprocessor controls a special network which samples signals of interest in the monitored system. The second processes the measurement data received from the first one and displays it. A special set of measurement instructions has been developed enabling the user to prepare easily programs to control the execution of complex measurements.
The architecture and software design of both microprocessors is introduced and several examples of performance measurements are presented illustrating the flexibility and usefulness of the designed monitor.

## 1. INTRODUCTION

It is now generally recognized that most computer systems operate at below maximum level of efficiency and that substantial cost savings are possible through effective performance management.

The effectiveness of a performance related decision varies according to the quality and precision of the input data to the decision making and verification process. Without an adequate means of obtaining this data, the improvement in performance cannot be precisely judged and potential cost savings will go unrealized.

Mathematical analysis might be inaccurate or even impossible due to the complexity of large computer systems. The enormous number of variables that must be considered, makes the mathematical analysis a tedious work and inadequate for this kind of problem.

More accurate tools for performance evaluation of operational computer systems are the hardware and software monitors [1-3]. There are several important advantages of a hardware monitor (HM) over a software one. The HM does not result in an overhead to the monitored system since its operation is transparent to the observed system. In contrast, a software monitor may consume 5% to 20% of the system resources [2] (e.g., CPU and channel time) and thus degrade the system performance and produce questionable results. In addition, a software monitor must be specifically designed for the particular computer system and the operating system in use, while a hardware monitor is portable and its operation is independent of the particular operating system. Some system performance parameters cannot be accurately measured by software techniques while they pose no problem to the HM with its high data rate. Examples are: internal CPU work, memory use, I/O interface activity and control unit or drive activity.

Measuring the behavior of complex systems such as multi-processors or multi-computer systems is difficult if not impossible by software tools. The HM has the ability of monitoring the parallel work of all the CPU's without being affected by the work of any of them.

A number of hardware monitors are offered commercially. However, after examining the available monitors the decision was made to design a more advanced and flexible one. This microprocessor-based HM has to be programmable and enable interactive operation for on-line analysis of system performance parameters.

The HM was designed to meet large computer system's requirements. An example of such a computer system is shown in Fig. 1. These requirements include monitoring of single/multi processors, slow/fast I/O processors and devices, shared I/O units, communication channels, etc.

The HM must be able to measure a large quantity of slow/fast signals, and be capable to display, print and store processed measurements. Another important requirement is to enable the user to prepare and run measurement programs for analysis of complex situations.

Some other unique features of the designed HM are pointed out in the following sections.

## 2. BASIC ARCHITECTURE

The HM that was designed has two major units, a Data Collecting Unit (DCU) and a Processing and Display Unit (PDU), each controlled by a separate microprocessor. The basic architecture of the system is described in Fig. 2.

The DCU contains a microprocessor that controls a special hardware network which samples and counts input signals. This hardware is fast enough to measure accurately signals up to 20 MHz. The input signals are obtained through probes which are connected to points of interest in the monitored system. Up to 256 such probes are connected to the DCU through a logic panel which enables the user to combine several input signals to form new ones of special interest.

The DCU collects the measurement data and converts it into a standard form. All the measurements are done in fixed intervals varied by the operator. It is possible to measure activity, time intervals, and count pulses or events.

The PDU is a general purpose microcomputer. It processes and displays the information sent, at each interval by the DCU. The display is updated every interval or retained on the CRT for any period of time. Using the monitor's peripheral devices, data may be

(1) recorded on floppy disk for playback and/or post processing by an analysis and report generation program.

(2) printed on a line printer, when hard copy is needed.

The user controls the monitor operation via an alphanumeric keyboard and the functions are defined using preformated CRT frames.

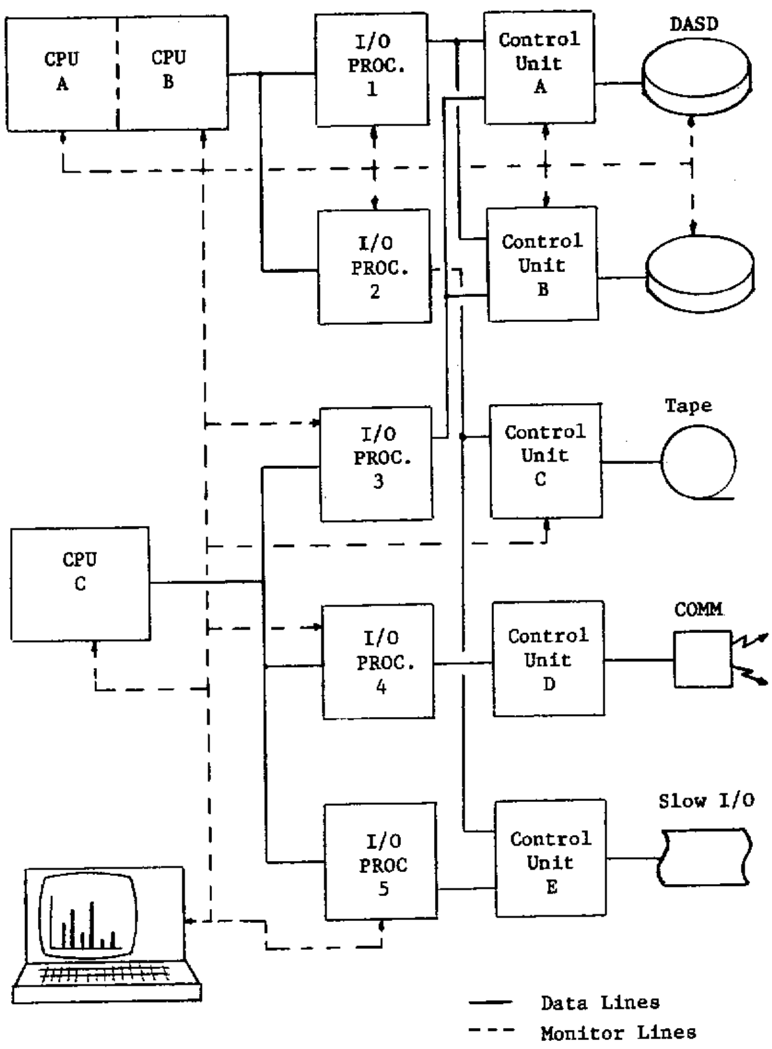


Fig. 1: A hardware monitor in a large computer system.

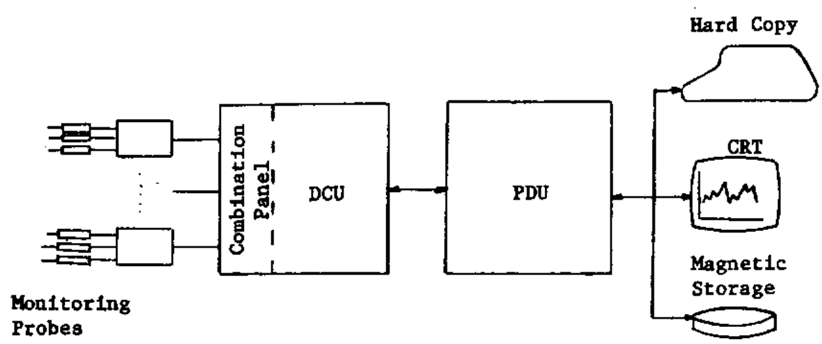


Fig. 2: The basic architecture of the hardware monitor.

## 3. DATA COLLECTING UNIT

### Hardware Structure

The DCU consists of a Z-80A microprocessor, measurement hardware and combination panel. Its block diagram is shown in Fig. 3. The DCU sensors can monitor signals produced by almost any logic family including ECL, TTL and other special types. The maximum distance between each sensor and the DCU is 150ft.

Up to 256 input signals from the monitored computer system are connected to the DCU through a combination panel. This panel produces 64 output signals to be monitored by the DCU. The 256 input signals may be combined within the combination panel to form new signals using a set of interchangeable modules. These modules include standard IC's like logic gates, encoders, decoders and specially designed modules. Examples of the latter are: (1) A sequence detector which receives up to five input signals and detects a predetermined (by the user) sequence, (2) A testing module producing various known signal sequences which are used to test the DCU's operation.

Sixty outputs of the combination panel are wired to the DCU while the other four can be selected under program control out of all 256 signals. These four signals may also be combined to form any logic function of four variables which is determined by the user interactively. Such a dynamic generation of logic functions is not possible in most existing hardware monitors.

The measurement hardware consists of a sampling logic and a set of 64 counters (16 bit each) that carry out the major part of measurements

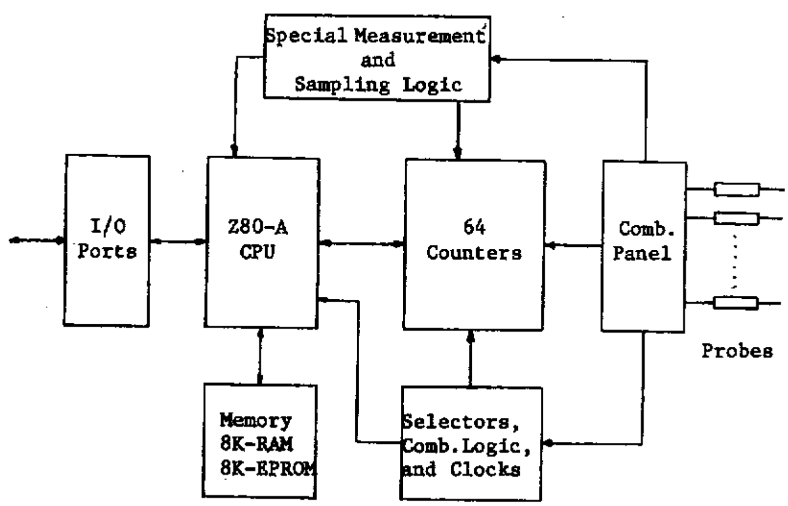


Fig. 3: Block diagram of the Data Collecting Unit (DCU).

done by the monitor. The counting is not limited to 16 bit since overflows of the hardware counters are accumulated by the software. The basic sampling rate is 2 MHz but can be increased up to 20 MHz. The sampling clock can also be supplied from an external source. The special measurement logic includes hardware for measuring time intervals (between 200 nsec and 99 sec), momentary states and word mappings. These types of measurements will be explained in section 5.

### Software Design

The DCU software controls the execution of measurements and the transfer of results to the PDU. The software can be divided into schematic blocks as shown in Fig. 4. The main program routine interfaces between the PDU and DCU, and controls the overall operation of the DCU. The DCU executes a special set of high level instructions designed for measurement purposes. These virtual instructions sent by the PDU are decoded by the main program in the DCU and executed by a set of instruction execution subroutines.

The program monitor executes measurement programs (of high level instructions), thus minimizing the system overhead which might be caused by sending single instructions.

The measurement monitor monitors the executions of a variety of measurements. It has been efficiently designed since its operating speed determines the maximum measurement rate.

The diagnostic monitor tests all the DCU hardware and it can isolate a single failing card. It insures that the DCU is operating properly and includes integrity test of EPROM's and RAM's.

## 4. PROCESSING AND DISPLAY UNIT

The PDU (an S-100 bus microcomputer) controls the collection, processing, buffering and outputting of all measurement data. It is composed of a Z-80A microprocessor and standard peripheral units. A block diagram of the PDU is shown in Fig. 5. A CRT and an alphanumeric keyboard are the interface available to the operator for controlling monitor operations and for the display of all measurements.

Two out of the four diskette drives are reserved for data recording, while the other two are reserved for the signals data base and the monitor software. Two asynchronous serial ports can be connected to modems allowing remote terminals to be connected to the monitor.

### Software

The major part of the PDU software is a dedicated software package designed to meet the specific requirements of the system. It uses standard I/O and file management routines which are part of the CPM 2.2 operating system. About 85% of the software was written in BASIC-80 programming language and only 15% are assembly language subroutines, including interrupt routines, DCU interface, diagnostic programs and peripheral drivers.

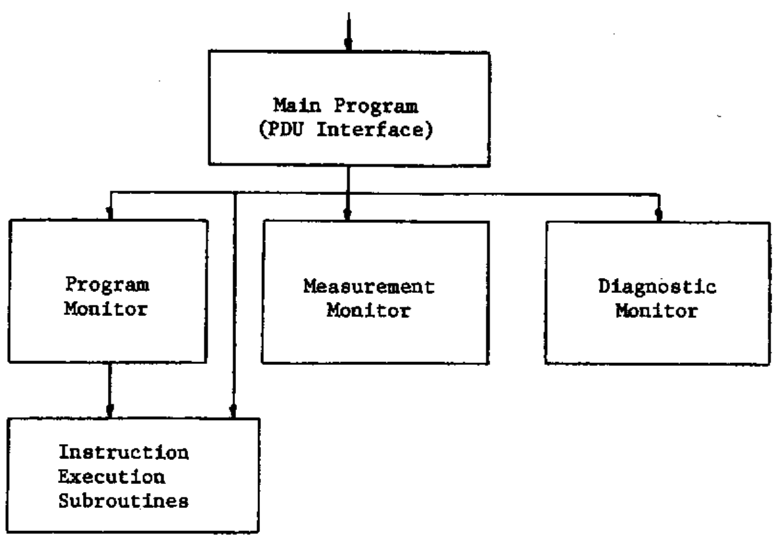


Fig. 4: The structure of the DCU software.

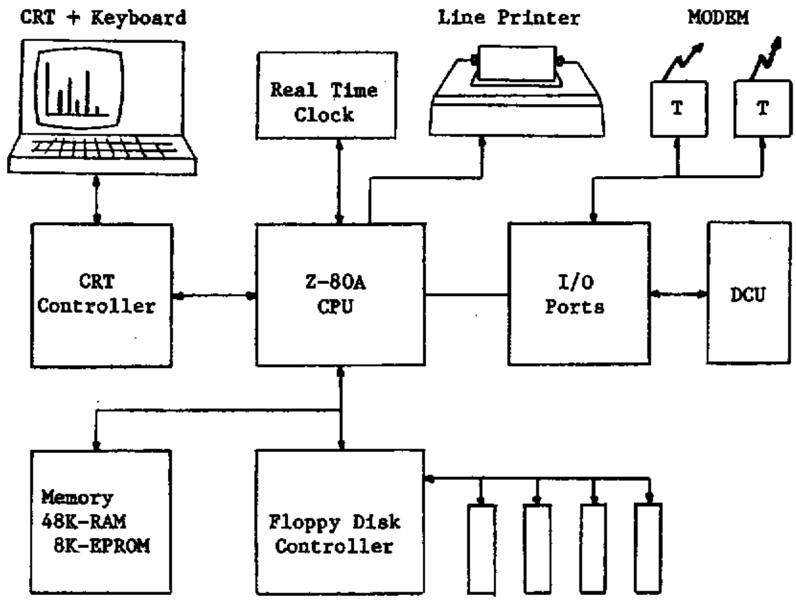


Fig. 5: Block diagram of the Processing and Display Unit (PDU).

The block diagram in Fig. 6 illustrates the major modules of the PDU software. The monitor control program is the main routine that coordinates between all other modules. It decodes operator commands, displays CRT frames and manages all disk file accesses.

The measurement program collects and processes the measurement data transmitted by the DCU. The data processing includes minimum, maximum and averages calculations. A software automatic gain control feature determines adaptively the scale factor of the graph's y-axis for maximum readability of the displayed graphs, according to the changes in the monitored signal.

It is possible to enter measurement programs written by the user in a special high level language and those programs are interpreted and executed by a user program monitor. This feature will be detailed in the next section.

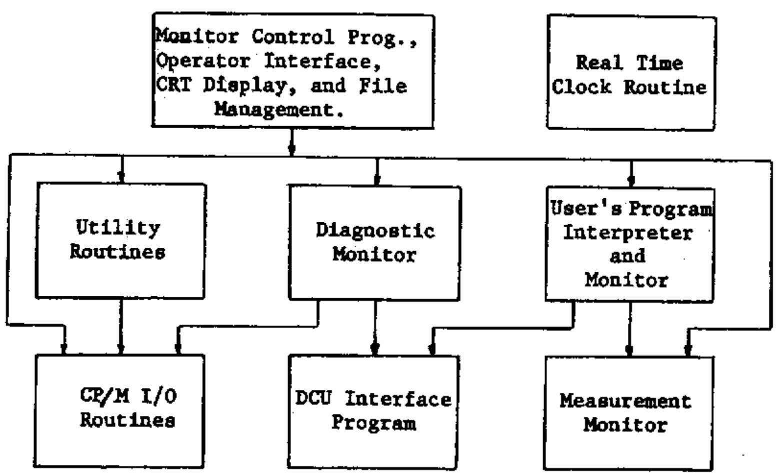


Fig. 6: PDU software.

### DCU - PDU Interface

A set of four 8-bit parallel I/O ports constitute the interface between the DCU and PDU. The PDU and DCU relationship is a master-slave one resulting in a very simple and straightforward communication protocol that insures high reliability and flexibility for the PDU-DCU link. Consequently, the DCU can be connected to any standard microcomputer that has parallel I/O ports. It is also possible (with only a few changes) to adapt the interface to serial ports with some degradation in the performance of the hardware monitor.

The virtual instructions originated by the PDU and executed by the DCU enable transmission of data and programs from PDU to DCU, configuring of signals, initiation of measurements and transmission of measurement data from DCU to PDU. Most of these instructions are 8 bit long and only few are 16 bit long.

## 5. CRT FRAMES

The monitor operation is controlled by the operator through a set of preformated CRT frames that allow simple and convenient activation of all monitor functions.

There are 16 frames which may be partitioned into two general classes. (1) Display frames (2) Control and service frames.

### Display frames

(a) Activity display: This display frame contains up to 16 vertical bars showing the activity of selected signals. The activity of a signal is either the percentage of time that a signal is active during the time interval or the number of times the signal is active during the interval. The display is updated at the end of every interval.

(b) History display: A graph showing the activity of a selected signal is displayed as a function of time. Every time interval a dot is added to the graph. Statistical values of the signal are also displayed and updated every time interval.

(c) State display: Momentary states of 32 lines are displayed after the DCU is triggered. The trigger source may be an external signal, the user keyboard or a program in the PDU.

(d) Spectrum display: This frame can display two basic measurements,

  (1) Time distribution (histogram)
  A signal received from the logic panel is timed and the duration time of each pulse is classified into one of 32 consecutive time ranges. At the end of the interval, 32 vertical bars are displayed, showing the number of pulses for each range. The size of the time range can be varied by the operator from 1 $\mu$sec to 10 sec.
  It is also possible to measure time between two pulses of different sources, and display it using the same procedure.

  (2) Word map
  Thirty two vertical bars are displayed corresponding to the 32 combinations of five bits. Each bar shows the number of times the corresponding binary combination occurred during the specified time interval. This measurement allows sampling of data busses or special signal groups (e.g. command code, address register, etc).

A display frame of any type which is currently displayed on the CRT can be printed and/or recorded.

### Control and Service Frames

The control and service frames include signal configuration frames, display configuration frames and special frames.

(a) Signal configuration frames
A set of three frames allows the operator to build a signal library. He can define new signals using existing ones and keep them for future use. For every new signal that is defined a file management program checks for proper configuration and validity prior to measurements. This is another unique feature of the presented HM.

(b) Display configuration frames
Associated with each display frame there is a display configuration frame used to define the signals to be measured and displayed by the corresponding display frame. Those signals were configured previously using the appropriate signal configuration frame and stored in the signal library.
For display configuration the user needs only to indicate the signals to be measured and the system will determine the way to measure them. Self explained CRT messages are generated in response to operator errors.

(c) Special frames
The special frame group includes :

  (1) Utility frame

  (2) Playback frame for search and display of prerecorded data.

  (3) A diagnostic frame for overall testing of monitor functions.
  The system is self-tested, and appropriate messages are displayed or printed allowing fast detection of faulty cards for minimum downtime.

(4) A program frame through which the operator enters and runs programs in a high level language especially developed for measurements. The language statements include all the monitor operations, I/O commands, mathematical operations, etc. The use of this high level language is extremely convenient when writing programs to analyze complex situations, where static displays or batch reports are not useful.

As an example of the usefulness of the measurements language consider the following situation. Deadlocks in large computer systems may be the result of a complicated sequence of events. Such a sequence may happen in a relatively short period of time, not allowing the system engineer to isolate the reasons that resulted in the deadlock. To determine the causes of the deadlock the system engineer may prepare an appropriate measurement program for monitoring sequentially a set of relevant signals. Deviation of a signal from its desired value (or duration) may result in a branch to another subroutine designed to monitor another set of signals and so on.

The monitor can record the sequence of events for later analysis and it can also be programmed to send appropriate messages, for fast detection of hazardous situations. The monitor can also be programmed to execute measurements requested by remote terminals.

## 6. PERFORMANCE MEASUREMENTS

Various system performance measures can be evaluated and analyzed using the hardware monitor. Performance improvement can be achieved by optimizing the use of system resources. Possible improvements are in configuration management, equipment utilization, operating system and application programs runtime, data base management, etc. The following examples show how the HM may be used to achieve these goals.

Configuration Management

(1) Balancing the use of system resources to increase throughput and eliminate bottlenecks.

(2) Objective selection of equipment and validation of vendor's specifications.

Software

(1) Evaluation of tradeoffs between main memory size and improved performance.

(2) Identification of application programs which utilize heavily critical computer resources and detection of program sections that their optimization can improve system performance and reduce cost.

Data Base Management

(1) Elimination or reduction of underutilized resident file space.

(2) Better allocation of disk files for minimum arm movements.

The following example shows how the hardware monitor can be used to improve performance through configuration management. One of the main reasons for performance degradation in computer systems is contention for the use of a resource (in particular shared disks and control units) . The major question is to decide whether or not the current level of contention is degrading service, and if so, how can it be reduced. Several measurements can be made by the HM to analyze this problem. For example, suppose that the monitored system consists of a dual CPU with shared peripheral units as shown in Fig.7.

The following measurements can be performed:

(1) Activity of the following signals (activity display).

| | |
|---|---|
| a) CPU 1 busy | f) CH 2 (CPU 2) busy |
| b) CPU 2 busy | g) CTRL A busy |
| c) CH 1 (CPU 1) busy | h) CTRL B busy |
| d) CH 2 (CPU 1) busy | i) CTRL C busy |
| e) CH 1 (CPU 2) busy | j) CTRL D busy |

A measurement session as described above can detect poor utilization of CPU 1 due to extensive use of shared disks by CPU 2. This fact can be detected by correlating the wait signal of CPU 1 with the busy signals of the CH's and CTRL in the other CPU.

A measurement session as described above can detect poor utilization of CPU 1 due to extensive use of shared disks by CPU 2. This fact can be detected by correlating the wait signal of CPU 1 with the busy signals of the CH's and CTRL in the other CPU.

(2) Waiting time for disk service at each CPU (spectrum display).
This is the time elapsed between the I/O request made by the CPU and the initiation of the data transfer. Monitoring these two events using a spectrum display, (in the range 0-320 msec with a 10 msec resolution) enables the evaluation of the service time and the number of interrupts. More than three missed interrupts for most of I/O operations can be caused by a very busy control unit and it can explain poor CPU performance.

(3) Paging activity as function of time (history display).
It is possible to count the number of I/O operations as a function of time for a paging disk. Excessive paging transactions can be the result of a too small physical address space or software problems. The graph in a history display can show peaks of paging activity allowing the system manager to correlate those peaks with system activity.

(4) Main/alternate path utilization (spectrum display).
For computer systems that have two access paths to I/O devices, it is desired to improve the main/alternate path utilization. Using word map type measurements it is possible to display for a disk string, the number of I/O operations for every disk from both paths. This measurement may help to decide whether the two paths are necessary and what are the tradeoffs for eliminating one path.

## 7. CONCLUSIONS

The architecture and software design of an advanced hardware performance monitor have been presented. The hardware monitor is a dual-microprocessor system especially designed to measure the performance parameters of complex multi-processor systems. The unique features of the microprocessor-based hardware monitor have been introduced and examples illustrating its usefulness have been presented.

## 8. REFERENCES

[1] Ferrari, D., Computer Systems Performance Evaluation, (Prentice-Hall Inc., 1978).

[2] Sauer, C.H., and Chandy, K.M., Computer Systems Performance Modeling, (Prentice-Hall Inc., 1981).

[3] Fuller, S.H., Performance Evaluation, in, Introduction to Computer Architecture, Stone, H.S., (ed.), (SRA 1975).

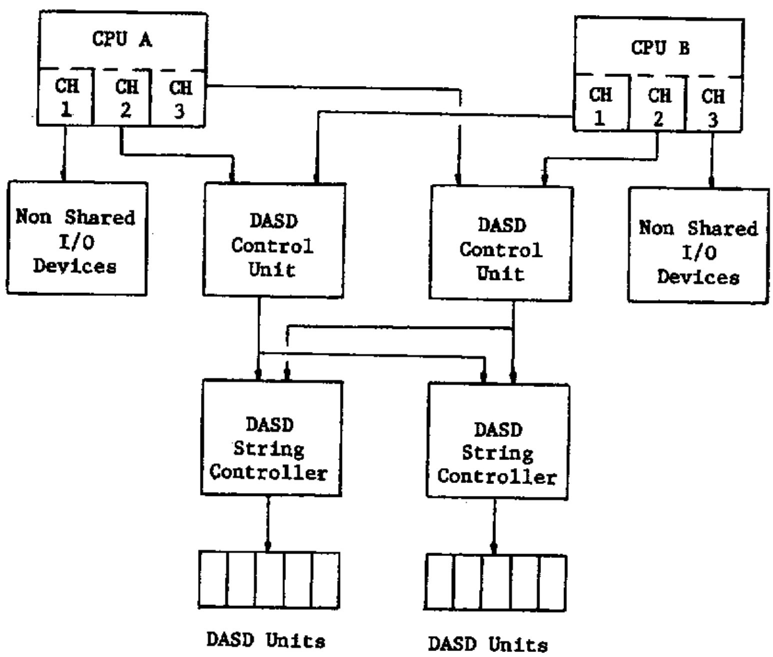


Fig. 7: A dual CPU system.